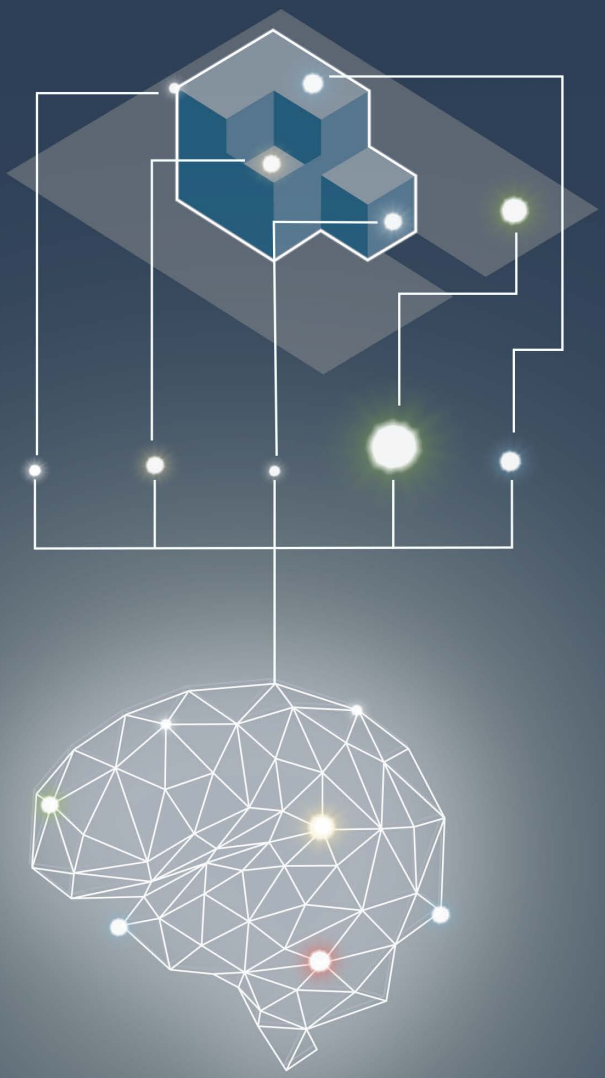
MSc. Thesis
Geomatics for the Built Environment

Re-thinking the region

Systematic evaluation of residential location choice under disaster risk



Kotryna Valečkaitė, 2018

TU Delft ARUP

RE-THINKING THE REGION

SYSTEMATIC EVALUATION OF RESIDENTIAL LOCATION CHOICE UNDER DISASTER RISK

A thesis submitted to the Delft University of Technology in partial fulfillment of the requirements for the degree of

Master of Science in Geomatics for the Built Environment

by

Kotryna Valečkaitė

November 2018

ISBN 978–94–6186–989–0

The work in this thesis was made in collaboration with the:

Chair of Design Informatics
Department of Architectural Engineering and Technology
Faculty of Architecture & the Built Environment
Delft University of Technology

ARUP

Arup
Design, Engineering and Business consultation firm
Amsterdam

Supervisors:	dr. ir. Pirouz Nourian	TU Delft
	dr. ir. Geertje Bekebrede	TU Delft
	dr. Michele Palmieri	Arup
	Alex Christodoulou	Arup

ABSTRACT

In this thesis we present a computational framework, which allows simulating residential location choice. The specific application of the model focuses on representing populations under disaster risk. The aim of the tool is to enable public planning agencies to explore the synthesized households' location choice under risk and different spatial, financial policy scenarios. The simulated urban system is represented as an agent-based model, where the households are the agents choosing between discrete options to relocate from one house to another. Their choice behavior is built upon a notion, that people make decisions based on regret. This is done with the help of Random Regret Minimization (RRM) model, allowing to capture varying levels of regret (profundity) and enabling incorporating multiple attributes of different dimensionality. Given the agent heterogeneity and the changing availability of the building stock, the choice sets are dynamic. Therefore, the traditional RRM approach would not return stable results: with every new option housing available it would have to be re-calibrated. As a solution, we propose re-interpreting the classical RRM model by scaling the beta values by the choice set attribute variance. This allows to represent β as a unit-less preference weight, associated with a homogeneous population group.

We apply the framework to two different scale case-studies in an earthquake-prone area in Groningen province, The Netherlands. The data used for simulation includes several public and private spatial datasets, as well as aggregate level statistical data to synthesize the properties of the households. As a showcase, we applied the simulation assuming homogeneous and equal preference weights for 7 optimization criteria. These criteria relate to static properties of the building stock and household-related dependencies to the network (job, school locations). To further exhibit model usability within public sector agencies, we also apply the model on several financial policy scenarios. The output of which is captured on both aggregate and semi-disaggregate levels, allowing for interactive exploration of the effects of the proposed scenarios. The model outcomes correspond to the expectations set prior to simulation. It showcases convergence, anticipated optimization behavior and spatial patterns, corresponding to the building stock properties of the region.

ACKNOWLEDGEMENTS

Firstly, I would like to thank all my supervisors, Pirouz Nourian & Geertje Bekebrede from TU Delft, and Michele Palmieri & Alex Christodoulou from Arup. Additionally, as well as thesis co-reader, who chose to remain anonymous. They gave me possibility to step away from my project and see it from new perspectives.

Moreover, I need to extend my gratitude to Arup Amsterdam in general. The company not only provided funding for my research, but also allowed realistic framing, the case-study, for the thesis. Apart from my company supervisors dr. Palmieri and Alex Christodoulou, I need to give extra thank you for Laurens Versluis, who was willing to discuss and give feedback on sometimes rather vague subjects or products. Additionally, I am grateful to Rinke Kluwer and Kubilay Hicyilmaz, who showed me how to make my presentations more approachable and interesting.

Furthermore, I have to thank my partner Marijn Tiggelman, who has been superhumanly patient during the hard periods. With his expertise in the field of mathematics, programming and numerical modeling he always had a good tip on how to circumvent the issues. Additionally, he is solely responsible for the success of the optimization and model execution times reasonable with the days I had left to run them. This work would have never reached its potential without your help!

Lastly, thank you to my family who has been supportive throughout my 5-year journey in TU Delft: it is amazing to always have somebody to talk to on the other end of the line, as well as the safety net to fall back on when the hard times hit.

CONTENTS

LIST OF FIGURES

LIST OF TABLES

List of Algorithms

ACRONYMS

ABM Agent-Based Model

ABMS Agent-Based Modeling and Simulation

AH Agent Heterogeneity

BAG Basisregistratie Adressen en Gebouwen

BGT Basisregistratie Grootschalige Topografie

BRK Basisregistratie Kadaster

CBS Centraal Bureau voor Statistiek

CTC Complexity Theories of Cities

CVW Centrum Veilig Wonen (Center for Safe Living)

DCM Discrete Choice Model(ing)

DES Discrete Event Simulation

DRM Disaster Risk Management

DF Data Frame

ETL Extract, Transform, Load

EV Extreme Value

FME Feature Manipulation Engine

FW Floyd-Warshall

GIS Geographic Information System

GEM Global Earthquake Model

GESU Groningen Earthquakes Structural Upgrading

id identification (number)

IPF Iterative Proportional Fitting

LAW Lange Afstands-Wandelpaden (Long Distance Walking Paths)

LISA Landelijk Informatiesysteem van Arbeidsplaatsen (National Information System of Workplaces)

LMR Land-market representation

LMS Landelijk Model Systeem (National Modeling System)

LUT Land-Use Transport

LUTI Land-Use and Transport Integrated

MADM Multi-Attribute Decision Making

MAS Multi-Agent Systems

MCDA Multi-Criteria Decision Analysis

MNL Multinomial Logit

MODM Multi-Objective Decision Making

MSM Microsimulation

NAM Nederlandse Aardolie Maatschappij (Dutch Petroleum Company)

NWB Nationaal Wegen Bestand (National Road Dataset)

NRM Nederlands Regionaal Model (Dutch regional model)

NP Nondeterministic Polynomial time

ODD Overview-Design-Details

OSM OpenStreetMap

PGA Peak Ground Acceleration

PS Public Sector

PSS Planning Support System

PC4 Post Code level 4

RP Ruimtelijkeplannen (spatial plans)])

RRM Random Regret Minimization

RUM Random Utility Maximization

SDSS Spatial Decision Support System

VBO Verblijfsobject

VV Verification & Validation

WFS Web Feature Service

WOZ Waarde Onroerende Zaken

1 INTRODUCTION

Models are theoretical abstractions, representing systems, such that their essential features and applications are identified [14]. This enables them to act as tools for experimentation, allowing testing theories in a controlled laboratory. In the context of urban modeling, they are essentially computer simulations (ibid.). Models digitally represent phenomena and enable testing of theories without them being implemented in the real world [14]. The first constructs of this kind appeared in the 1960s [167, 13, 5, 173]. They came as a response to the growing need to go beyond the abstract theoretical models, which had not direct value for informing Public Sector (PS) decision making processes [167]. According to Batty [14],since their advent, urban models have gone through series of changes Figure 1.1– starting from static and aggregate Land-Use Transport (LUT) model approach, and progressing to the more recent disaggregate and dynamic, Agent-Based Model (ABM) approach (more in Chapter 3). As the latter gained ground, the discourse moved away from the traditional role of models as tools for scenario testing, as it became nearly impossible to calibrate or validate them. Batty [14] further elaborates by stating, that due to this role of models shifted to frameworks for informing and structuring the formal and informal dialogs, allowing for participatory decision support.

1.1 WHY MODEL?

But why would one consider modeling a city or a region? When introduced into the decision-making process in a controlled way, models can bring scientific knowledge and structure to the planners or policy makers' table [159]. In fact, these decision makers constantly 'run' models. That is to say, they imagine specific aspects of urban dynamics in relation to their intervention. Epstein [54] calls these mental exercises *implicit* models– their assumptions are hidden, internal consistency untested, logical consequences and relation to data unknown. However, once these aspects are reversed a model becomes *explicit*, allowing others to replicate the results. However, the goals of a model often go beyond the often expected prediction. Epstein [54] identifies 16 additional reasons, including: explaining the phenomena, guiding data collection, illuminating core dynamics, educating general public and disciplin-

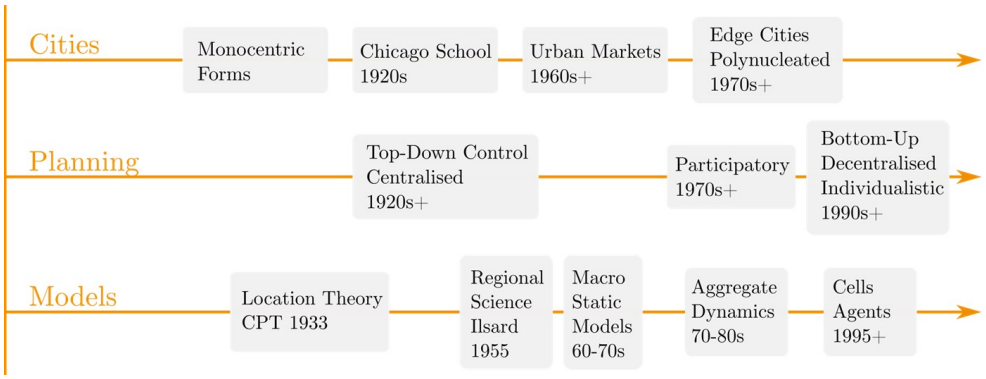


Figure 1.1: The intersecting time-line: from 19th to 21st century (based on [13])

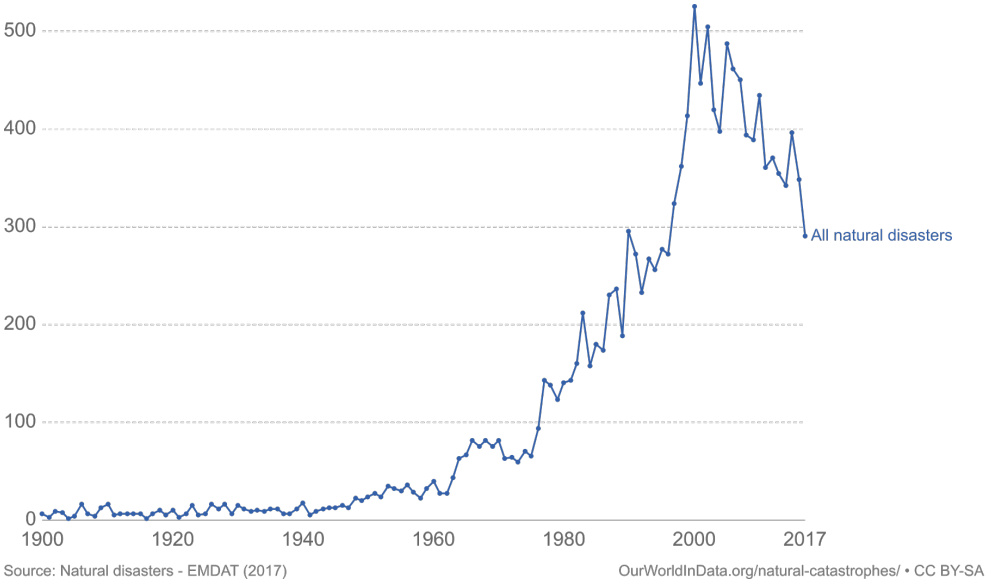


Figure 1.2: Number of reported natural disaster events per decade [138]

ing policy dialogue. When it comes to cities, Pumain and Reuillon [125] expand this list with the need of: (1) tools for large scale problem solving due to extreme urbanization; (2) keeping track of interactions between different urban systems, as they develop co-dependently; (3) exploration of regulation capabilities in self-organized dynamics to determine the plausible evolution and growth scenarios; (4) means for monitoring and evaluating the system for resilience and sustainability.

1.2 DISASTERS AND REGIONAL MODELS

Modeling also plays an important role in Disaster Risk Management (DRM). The main goal of this field is to minimize the impact of disasters by strengthening the population's coping capabilities and reducing the possibility of disasters occurring [24]. Within the disaster management cycle (Figure 1.3) models come into play in the preparation phase, allowing for understanding the processes of disasters and vulnerabilities [6]. Paradoxically, environmental impacts are rarely considered in general operational urban models [5]. In rare cases, that it is part of it (e.g. ILUMASS [168], TRESIS [75]), it is defined as a hindrance related to air or noise pollution. We also observe, that disaster risk is also not considered. Among many of these examples fall TIGRIS XL [178], an aggregate model used for land-use modeling in the Netherlands; microsimulation-based models PUMA [55] and UrbanSim [164] (more on the subject in Section 3.2). This is particularly important in the light of ever-increasing occurrence of natural disaster events (Figure 1.2). The data from International Disaster Database [138] shows, that even though the DRM efforts are bearing fruit (i.e. the casualty numbers have been diminishing), the damage costs have only been increasing. With this in mind, we see the urgent need to combine the knowledge of DRM with general urban and regional models. In this project we approach this gap, by focusing on one of the urban sub-systems- residential mobility.

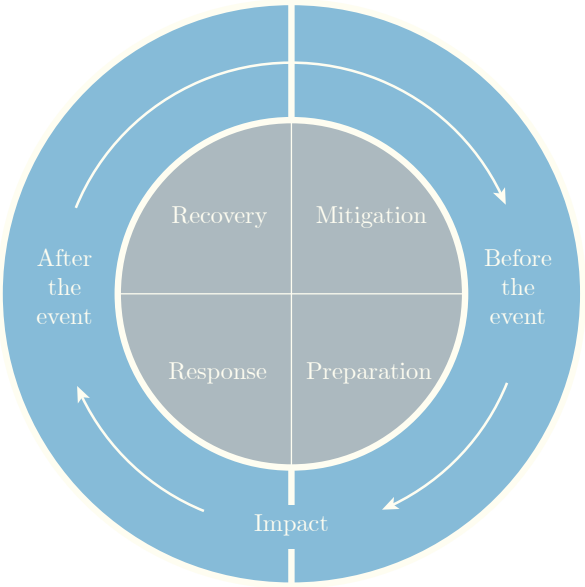


Figure 1.3: The disaster management cycle (adapted from [6])

1.3 FRAMING

Our project focuses on the smallest unit in urban environment connected to a residential building– a household. We are interested in their behavior relating to the residence choice, as this is one of the core mechanisms underlining the emergent patterns of cities [115, 93, 17, 4]. Within this perspective, we choose to inspect a single crisis situation– one point in time, when people are faced with an imminent risk, relating to their location and housing unit properties. This setting can be constructed by defining a system of households as heterogeneous agents, where they are faced with a variety of options to improve/optimize their living conditions. Namely, they pertain to households' perception of (1) residence properties (e.g. size, collapse risk), (2) location (e.g. proximity to school, job, amenities) and (3) social networks (e.g. socio-economic status of the vicinity, proximity to friends). Models dealing focusing on this part of urban system are called residential mobility and housing choice models. Thus this research strives to contribute to the field of urban modeling, by explicitly handling disaster risk as a variable influencing the behavior of the population of the area. Namely, risk becomes an attribute associated with a specific building in the region, as it not only depends on the location, but also the structural properties.

1.4 CHALLENGES

Housing choice and residential mobility models form an integral part of majority general urban models (see Section 3.2). Despite their long history [13]– Land-Use and Transport Integrated (LUTI) being the gold standard for regional planning [166]– they still face significant challenges. Most notably, many of the existent models are difficult to use and adapt [57, 165]. This can be attributed both to the lack of transparency [19, 118, 124, 165] and validation [142, 165], and to the ad-hoc nature of modeling solutions [70, 13]. This plays a role in the fact, that many of the operational models are psychologically inaccurate [38, 165]. Another key point is that large number of models are still grounded in the aggregate traditional urban economics and gravity approaches, rather than disaggregate, focused on the behavior of individuals [13, 5]. The latter approach, allows one to explore new dynamics

and transitions in complex, self-organizing systems [78]. This comes at the cost of computational complexity and large requirements for data [165, 14]. All of these issues block wide-spread adoption of the integrated models within the PS planning organizations [163, 63, 159, 151].

1.5 SCOPE AND RESEARCH QUESTIONS

In this project we primarily strive to contribute to the field of urban modeling, by explicitly handling disaster risk as a variable influencing the behavior of the population of an area. Nonetheless, this thesis also addresses each of the challenges presented, with the goal to create a computational framework for a *usable* tool. By usability we imply, that it should be easy to adapt and expand, be transparent and show the capabilities to answer questions posed by the focus user. In our case the focus user is Public Sector (PS) organizations. Thus, conscious of this duality in perspectives, we formulate the goal and questions of the research as following:

OBJECTIVE To define a *usable* computational framework, allowing for large scale spatial and disaggregate simulation of household location choice behavior under disaster risk.

MAIN RESEARCH QUESTION How to build a computational framework examining the residential choice behavior of households within a regional, disaster situation, given public sector agency-defined policy scenarios?

SUB-QUESTIONS

- How to abstract housing choice behavior of households in a disaster situation on a regional scale?
- What modeling approach would be suitable for such abstraction, given a PS planning or policy making process?
- What (type of) data could be used to generate model optimization criteria values and what is their relation to the data available for the case-study?
- What output should the model have and how to communicate it to the PS organization given a disaster mitigation or preparation situation?
- What are the uncertainties and limitations of the model how to circumvent them?
- What type of data would be necessary in order to calibrate such a model?

The research case-study area is in Groningen, the Netherlands. The region is affected by human-induced seismicity [160], whilst the building stock largely consists of structures with no anti-seismic regulations ([10] in [116]). This creates a seismic risk, paving way for a large scale (around 250.000 affected buildings) structural upgrading project. Given the thesis scope, we focus on a subset of the area with around 1000 households and 1400 building nodes.

The focus group of the show-case tool– PS organizations– are chosen due the available access to PS consultants. The framework for residential mobility is created to answer questions specific for this group, but due to time limitations only allows for pre-defined input or 'what-if' scenarios. Other deliverables and features are summarized using a simplified version of the MoSCoW methodology [51]. The minimal product requirements are mentioned within "Must". "Should" paragraph documents highly wanted features, whilst what falls beyond the scope is covered in the "Could and Won't" sections.

WITHIN SCOPE – MUST Research existing models and modeling techniques; inventory, evaluate and process data; research and define model attributes, bringing special attention to the aspect of risk; identify model uncertainties; create at least 2 'what-if' scenarios, which would allow to explore the questions posed by PS organizations; define the data collection requirements for calibration; identify the algorithm performance in the context of near-real-time usability.

WITHIN SCOPE – SHOULD Create a proof-of-concept tool showcasing the usability of the framework; validate the framework usability with a series of semi-structured interviews; test the model on different spatial scales; implement a dynamic set of rules to capture the mechanism of structural upgrading.

OUTSIDE SCOPE – COULD AND WON'T Create a predictive model; collect the real behavior data and calibrate the model; create a general urban model or integrate it in an existing framework; optimize the code for near-real-time usability.

1.6 READING GUIDE

The thesis document is organized as follows:

- Chapter 2 presents formulation of the problem, which helps us introduce the basic terms. This is followed by showcasing the complexity of the problem in computational terms. We conclude the chapter by providing argumentation for model constraints given a problem framework from the field of location science.
- Chapter 3 presents a selection of related works, that helped us define the model framework. Three major fields of studies are covered: spatial decision and planning support tools, urban and regional models and discrete choice models.
- Chapter 4 explains the methodology. Here you can find the data structure, pseudo-codes and algorithms. Additionally, the chapter discusses the Verification & Validation (VV) methods, as well as procedures for usability validation.
- Chapter 5 delves into implementation details, covering subjects of data sources, collection and processing, software choice, including code performance and optimization issues.
- Chapter 6 gives an overview of the model and VV outcomes. This section of work also includes a discussion, which covers the requirements and uncertainties of the model.
- Chapter 7 concludes the research, discusses the results and lists future recommendations.

2 PROBLEM DEFINITION

In this chapter we delve into the details of the definition and complexity of the phenomenon at hand. We begin with explaining this from the perspective of a hypothetical or a 'toy' problem, in the manner of Batty [16]. Then we define the basic mathematical representation and consequently on what it means in computational terms. As last we discuss the specifics of the building stock and population in the research area, based on which we can create simplified aspect representations for the model.

2.1 HYPOTHETICAL PROBLEM

We explain the problem, firstly, as a hypothetical or a "toy" problem (Figure 1). In this case, we focus only on finding the best locations for two households. They can either stay in the current residence (in blue, 2.1) or change their location (marked in yellow, 2.1). If household 1 chooses one of the alternatives, household 2 is excluded from attaining it. These location-bound choices can be described as a collection of attributes (a_n, 2.2) or a vector, based on which a decision can be made. Both households in this toy problem would have a unique feature vector. This is due to that household 1 only has one job relation to the network, whilst household 2 has job relations (blue links, 2.1-2.3). These location relations are identified by attribute a_j or the travel time to work (2.3), where each link cost 1 unit of time to traverse to each location. This attribute inexplicitly captures the size of the working population as well. The travel times to each work location are not averaged, but simply added together. Additionally, each household's access to a closest shopping facility (a_s) is assessed with the same unit and way as a_j, where the travel time is multiplied by the household size.

A scenario in our case is represented by a $1 \times n$ feature vector (2.2), or, if we look only at a_j and a_s, as a 1×2 vector. Consequently, each of the scenarios can be represented as a column in a matrix or a data point in a R^n space, where n corresponds to the amount of criteria based on which we evaluate an option (2.3).

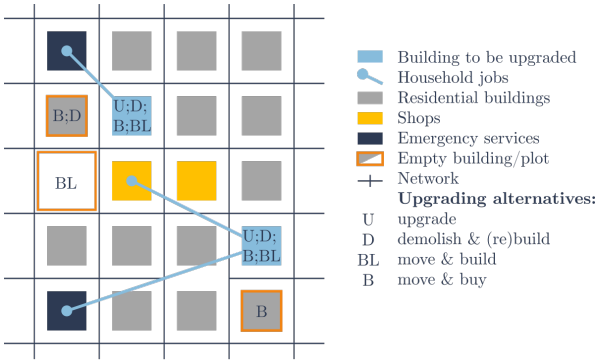


Figure 2.1: Toy problem: top left building to be upgraded (in blue) is referred to as household 1, the other- household 2 (own archive)

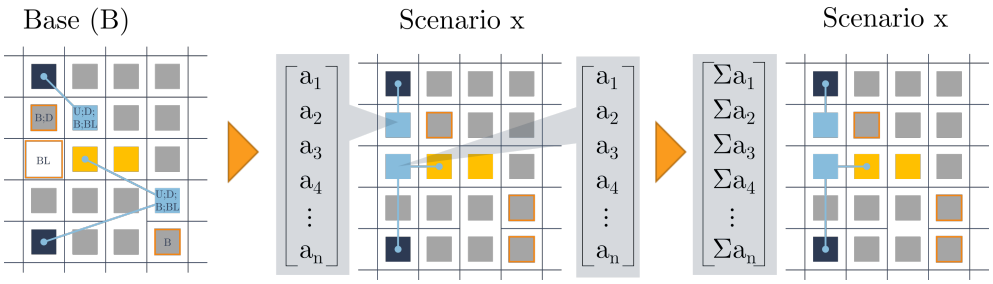


Figure 2.2: From the base to an alternative scenario– each household evaluates a residential location on a set of attributes, which are collected in a vector. A scenario is identified as a summation of all of those vectors

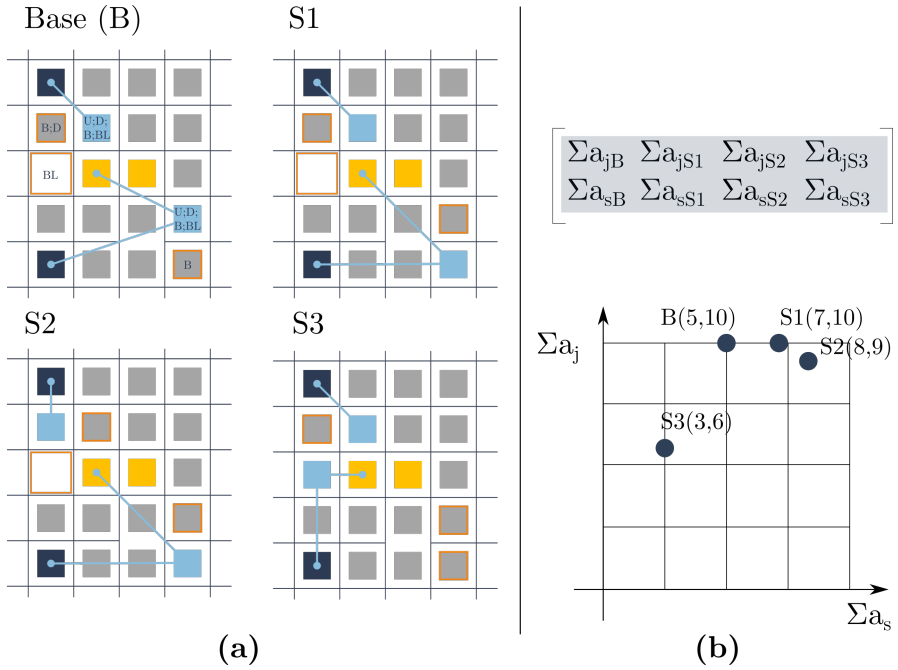


Figure 2.3: Starting and three alternative situations, scenarios (S1-3), compared on aggregate household access to jobs (a_j) and shops (a_s)

If we only evaluate 2 criteria and 3 alternatives (2.3), the decision can be easily made by visualizing it as a scatter plot. In the exemplary case, we care to minimize the travel times to shops and job locations (a_j and a_s). In Figure 2.3 sub-figure a, we see 4 scenarios where the household can potentially move. In sub-figure b, the same 4 scenarios are shown as points in a two-dimensional solution space. One scenario (Figure 2.3, Scenario 3 (S3)) stands out from the many, as it is less in each dimension (attribute) than any other alternative. However, if one would generate more alternatives (2.4), it is likely, that rather than finding one "best" solution, a set of solutions would be found, where one of the attributes cannot be further improved without deteriorating the other. This type of solution is called a Pareto optimal and their collection is a Pareto front. Any of the alternatives on this front can be identified as equally "good" and thus it is up to user to define the best solution, based on their personal preference.

2.2 DEFINITIONS

The problem as we described above can be summarized as the following:

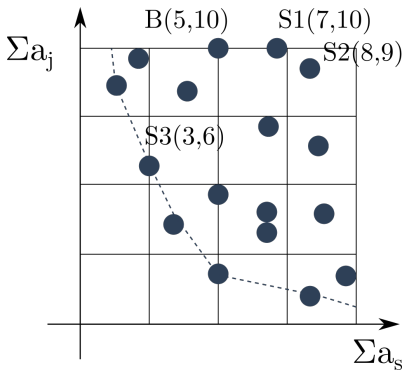


Figure 2.4: Hypothetical Pareto front (dashed line) of the "toy" problem, expanded with imaginary alternatives

- every household location h has a cost vector (e.g travel time to a facility, intervention cost), with number of locations h being l, such that $h \in \{1, 2, ..., l\}$;
- each cost type c in the cost vector is a criterion to be minimized, with number of cost types c being m, such that $c \in \{1, 2, ..., m\}$;
- a scenario i is a unique permutation of all l households' at available locations d, represented by a collection of l households' cost vectors, with number of scenarios i being n and number of available locations d being k, such that $i \in \{1, 2, ..., n\}$ and $d \in \{1, 2, ..., k\}$;
- to compare scenarios, we aggregate the households cost vectors per cost type c: $C^c_{i,h} = f(i, h)$
- the number of aggregate cost vector j corresponds to the number of scenarios i, so $i, j \in \{1, 2, ..., n\}$
- a scenario cost vector j_x is (globally) Pareto optimal if there is not another scenario cost vector j_y, such that $f_m(j_y) \leq f_m(j_x)$ for all m and $f_p(j_y) < f_p(j_x)$ for at least one index p (for more see [104, 105])

2.3 COMPUTATIONAL SCOPE

Returning to the project's case-study, as of December 2017, there were around 6000 active building portfolios within the highest hazard zone (Figure 2.5), each corresponding to one household. Assuming, that each household can only stay or exchange their living location to another affected building would lead to 6.000! different scenarios or location combinations. If we take into account the vacant buildings in the area, 4.07% as of 2016 (based on municipal data from CBS and the municipality coverage within the highest hazard zone, Figure 2.5), we have 1260 vacant residential buildings. Within the 'toy' problem, we assume that aech household has 5 choices associated with a single location: (1) accept the risk and do nothing, (2) structurally upgrade the house, (3) demolish the house and rebuild new, (4) move to another location and buy, or (5) move to another location and build. The three first choices are associated with the initial household location, whilst the last two with a new location. This leads to the following scenario amount formulation:

Total number of permutations due to re-location

$$\binom{k}{l} = \frac{k!}{l!(k-l)!} \tag{2.1}$$

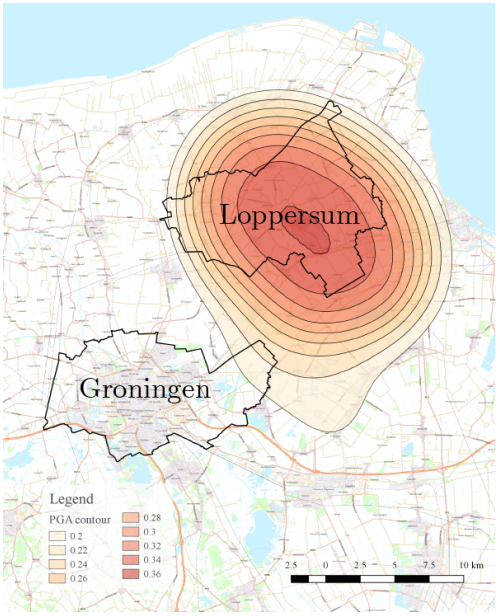


Figure 2.5: The municipal boundaries and the highest hazard zone with corresponding PGA contours (own archive)

Number of permutations (scenarios) with households staying in the original location

$$n_{hb} = 1 + l(k-l) + \sum_{p=1}^{P} \sum_{r=0} p((l-r)(k-l+p-r+1)) \tag{2.2}$$

Total number of permutations (location and upgrading choice included)

$$n = 2 \times (\binom{k}{l} - n_{hb}) + 3 \times n_{hb} \tag{2.3}$$

Where:
l – number of original household locations
k – number of available relocation possibilities (sum of l and vacant buildings)
P – upper bound of the sum, P=l

The resulting number of scenarios or all permutations is $7.04E21518$. This number would increase even further, if the empty parcels are added as an option for households (in this case, only one available) or the prognosis for the portfolio growth would be taken into account. This amount of options is far beyond what a human mind (see [106]) can evaluate. Additionally, even computationally, this problem is intractable (i.e. takes more than a lifetime to solve). More specifically, with each new criteria the algorithm scales linearly, but with each new household and building data point quadratically– $O(n^2)$. However, it must be noted, that many of the household and building permutations are far from realistic or satisfactory to the user. Therefore, the major challenge of this research is transparently reducing the number of permutations or finding the solution space boundaries, so that only the most adequate, human perceivable subset is presented to the user of the developed tool.

2.4 FRAMING

Given the context of the problem we can further reduce the scope by inspecting it from the location science perspective. This field primarily concerns itself with location-optimization of public or private sector facilities, rather than houses. Nonetheless, the publications on location problems relate to many research disciplines, such as economics, applied mathematics, computer science, geography and management [92]. We can observe obvious similarities with residential and facility location choice problematics. Both of them deal with finding optimal locations in space and thus share common problems while defining the modeling scope. Therefore, we decompose the residential location problematic on the basis of the framework proposed by Karabay (Table 2.1).

First thing to consider is the representation of space. The decisions made by households relate to discrete locations (parcels and houses), which are inter-related by a network of streets and roads. This suggests a mixed approach: the network can be used to define the properties of these discrete locations, whilst the simulation is run on the basis of discrete space.

The objective function approach from the household perspective primarily relates to the minisum problem. In other words, they try to minimize the travel times or distance to specific destinations in space (e.g. workplaces, schools, health care institutions, etc.). The minimax or p-center problem [29] in this perspective could also come in play as households could strive to minimize the longest travel time to a selection of destinations. This is relevant, as minisum tends to favor clustered destinations over the ones that are dispersed. In facility terms, this is primarily suitable for public services like ambulances or fire brigades, which reduces the negative impact on remote and poorest served clients. Lastly, the coverage [62] problems do not have direct application to residential location. These problems focus on finding a location, that would allow the service to reach the maximum amount of customers under a specific travel time or distance. Such as food delivery services, that serve customers only within specific postal code zones.

The solution method to our residential location problem cannot be solved by finding a single, optimal solution, as explained earlier in the chapter. Thus in this work, we have to focus on the heuristic approach and the methods relating to it.

The time horizon in our model should be considered as dynamic, as we cannot decide all the variables simultaneously. With that we mean, that the choice to relocate cannot be imposed on households. Therefore, the building vacancies and thus the options to relocate change through time.

The input parameters for residential mobility is likely to be a selection of both deterministic and stochastic variables, depending on which elements of the urban system we would be talking about. More specifically, anything related to personal information is most likely to be stochastic, whilst relating only to the built environment- deterministic.

The number of locations to be chosen for residential mobility is always one in our case. However, keeping in mind that we are simulating a system of many households, we are actually optimize the locations of many single facility choices at once.

The facility type aspect is not relevant within the scope of our research, as the housing choice in our model does not have negative or positive impacts on the other households in the area.

The last in the list, sector type, is private, as we are only choosing to inspect the privately owned building stock, which makes up the majority of the housing market in the case-study area [31]. Modeling more than one market would go beyond the time scope of the project, but should be considered as a future improvement.

Categorization Subject	Categorization Types	Explanation
Space	Continuous	In continuous models, demands are distributed continuously across a service region and facilities can be located anywhere in that region.
	Network	In network models, there is a network composed of links and nodes. Demands on nodes and facilities can be located on nodes or links.
	Discrete	In discrete location models, there are demands arising on nodes and facilities can be located only on a set of candidate nodes.
Objective function	Minisum	minimize average/total criteria values
	Minimax	minimize the maximum criteria values
	Coverage	maximize the coverage
Solution method	Exact algorithms	Algorithms that try to find the optimal solution.
	Heuristics	Algorithms that search for an approximate solution.
Time horizon	Static	Static models optimize the problem deciding all variables simultaneously.
	Dynamic	Dynamic models consider different time periods with data variation across these periods, and give solutions for each time period adapting to the different conditions.
Input parameters	Deterministic	In deterministic models, the parameters are forecast with specific values and thus the problems are simplified for easy and quick solutions.
	Stochastic	Stochastic/probabilistic location models capture the complexity inherent in real-world problems through probability distributions of random variables or considering a set of possible future scenarios for the uncertain parameters.
Number of facilities	One	The purpose of the problem is locating only one facility.
	Certain	Number of facilities to be located is a certain number.
	Uncertain	Number of facilities to be located is uncertain. Problem also searches for the number of facilities.
Facility type	Desirable	Closeness of facility (such as hospital) to demand center is better.
	Undesirable	People want these facilities (such as nuclear reactor) far from demand centers.
Sector type	Private	It seeks for maximizing profit while locating facilities.
	Public	Optimization of the population's access is the priority.

Table 2.1: Categories of facility location problems (based on [71, 89])

3 THEORETICAL BACKGROUND

> All models are wrong, but some are useful
>
> *G. E. P. Box and N. Draper* [26]

Based on the framing presented in the previous chapter, we will inspect the related concepts and put the model in a more general urban modeling context. However, we begin the story with some general considerations and applicability questions within the PS planning agencies.

3.1 POLICY–MAKING AND PLANNING

As mentioned in the introduction, models act as a laboratory for experimentation on urban fabric. In this section we will cover the types of interventions in which these experiments take place and when they play a role in policy making and planning processes. This section will conclude with descriptions of the tools, that are commonly utilized and the issues related to them.

Decisions within urban contexts

When talk about public agencies in our project we are referring to the fields of urban planning and policy-making. The first of the two primarily focuses on the design and regulation of space [58]. The second, as the name suggests, refers to a wider subject of formulating policies [30]. The term is usually used in political contexts in relation to, for instance, specific demographic groups, businesses or education in general. In both cases people deal with complex problems, which have inevitable reprecussions in space, as we will explain later in the chapter.

However, tn the most abstract sense, decisions in both of these disciplines can be of spatial or non-spatial nature. To illustrate them we draw inspiration from the field of transport modeling. Spatial decisions relate to adapting the street network, creating new parking facilities or public transport routes. Non-spatial decisions alter the properties of elements in the system: increasing transport tax, parking tariffs or public transport costs or introducing new travel mode. Of course the line between them is rather blurred. As for instance, specific pricing policies could be applied within spatial subsets.

'Wicked' problems

Both planning and policy making are fields dealing with 'wicked' problems [137]. As antithesis to 'tame', scientific problems, 'wicked' problems are ill-defined and ill-structured [137, 45]. For these problems we do not possess or are able to obtain all the necessary information. Moreover, they cannot be solved by exhaustive analysis, as there is no guaranteed 'correct' result. Cross [45] argues, that in such cases problem solvers tend to shift from problem-focused to solution-focused strategies. The same can be said about the goal urban planners and policy makers. One can go on analyzing the problem indefinitely, but their task is to come up with a solution.

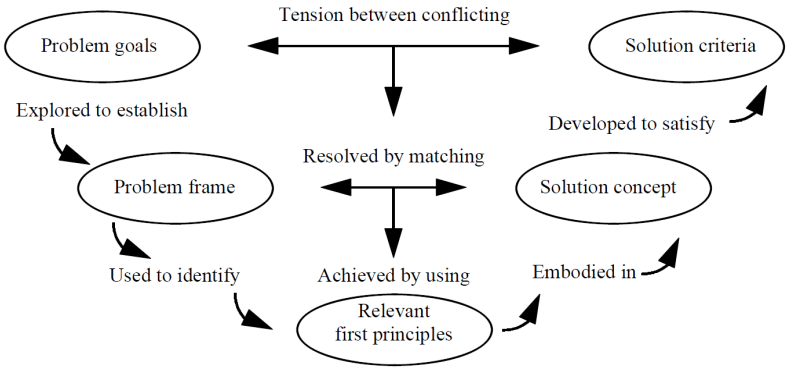


Figure 3.1: A general model of creative strategy [45]

Strategies

To deal with this, professionals tend to adapt a strategy presented in Figure 3.1, coming in line with the policy making process model shown in an article by Waddell and Ulfrarsson [167] and the planning process shown by Bovy et al. [25]. Firstly, one needs to define the goals of the problem. From the perspective of our research project, the PS organizations want to achieve a more effective (e.g. fast, less inconvenient) transition of households from unsafe to safe buildings. Secondly, this allows us to find the problem frame. In our case, it is finding interventions (e.g. financial policies, housing development), that would stimulate residential choice to upgrade or move and also positively impact their final choice. In the third step, one would start defining the first domain-specific principles, whilst in the fourth– to start creating the series of possible solutions. For our context, this means preparing series of hypothetical interventions (discussed in Section 4.4). The last stage is where one tests them based on specific performance criteria. At this point, urban simulations can play an essential role. They allow inspecting possible solutions in a structured manner and compare the impacts between the alternatives. Noteworthy, is that "there is no formal theory of scenarios and scenario analysis that tell us how to construct scenarios, how many scenarios to construct and how to reason between and across their outcomes " ([23], p.61).

Evaluation frameworks and tools

To start of, urban simulations are not the only tools used to support decision makers in urban contexts. For instance within sustainability perspective alone, Jensen and Elle [83] (also reinterpreted in [**?**]) identify four types of tools:

- Process (design) guides – provide frameworks to manage projects or policies (e.g. [42])
- Assessment methods – enable prioritizing and comparing different solutions by weighing specific aspects (i.e. criteria) (as in [10], general reviews of Multi-Criteria Decision Making in [59, 175, 162, 98])
- Monitoring (rating) tools – facilitate selection of indicators and benchmarks for formulating policies and monitoring their performance (e.g. [27, 156])
- Calculation tools – allow *ex ante* analysis of solutions, products and procedures (such as [164, 179])

Yet even with an ever growing interest and number of tools, little of them are utilized in practice [159]. Whilst focusing on simulations, the main obstacles have already been mentioned in Chapter 1. However, the literature on tools in planning processes indicate, that the information load of the simulation output is a significant

consideration. In essence, it is considered a type of decision task complexity, which in our case relates to the number of criteria and locations [82]. Namely, in a simulation working on a municipal or regional scale it would be far from understandable, if presented in disaggregate manner.

These issues are at the core of the research field of geo-visual analytics, which aims to support spatial decision making by 'augmenting' human capabilities with computational methods [122]. Batty et al. [18] state, that visual language is broadly recognized as easier to convey information than other forms of communication. Not only does it help to understand the outputs of a spatial model, but also in "getting the message across how the model actually works" [18]. This indicates, that we need to go beyond simple framework definition and output visualization in a static manner, as this would not communicate the full potential of the work. To do so we borrow the principles of enhancement through interactive exploration and discovery [122, 121]. In other words, the end user should be provided the possibility to explore and filter the outputs of the data, whilst also utilizing techniques as animation to fully showcase the temporal dimension of the residential mobility and bottom-up optimization.

3.2 URBAN AND REGIONAL SYSTEMS

Regardless of their issues, urban models, enable the testing of location theories against data and generating future locational patterns [14, 76]. This is especially interesting, when evaluating phenomena from a complex systems perspective.

A system is complex if its "elements interact and affect each other so that it is difficult to separate the behavior of individual elements" [65]. Important to note, that from this perspective urban environments are often seen as series of subsystems (e.g. residential, commercial, physical infrastructure), reflecting the location of human activity such as learning, working, living or shopping [4]. A model of a singular sub-system is called a "*partial model*", whilst a collection of them (two or more) are called "*general*" [12]. Each of these elements are related to each other, as illustrated by the 'land-use transport feedback cycle' [170]. The framework conceptualizes the nature of this as a two-way dynamic relationship between different sub-systems in the urban environment (Figure 3.2). According to this framework, the distribution of land use determines the location of activities and vice versa. This idea is embodied in the field of LUTI (reviewed in [165, 169]), where modeling households' residential choices is one of the greatest challenges [44]. From a time scale perspective, residential mobility falls in one of the slow processes within the system.

3.3 RESIDENTIAL LOCATION CHOICE MODELS

These types of models primarily focus on the household level probability of the move, and the likelihood to move to a specific location and building [84]. These decisions, which are by definition spatial, are linked to others such as car ownership, job and school choice. However, even every-day behavior, like what travel mode to choose and where to go shopping, might have influence on residential location choice. Each of these have their corresponding transaction costs, relating to the importance of the stakes associated with each decision. For instance, changing mode of travel due to traffic congestion would not likely have any impact on the future of the household or an individual. However, choice to switch to a different school or change of job, might. The core questions are discussed in a working paper

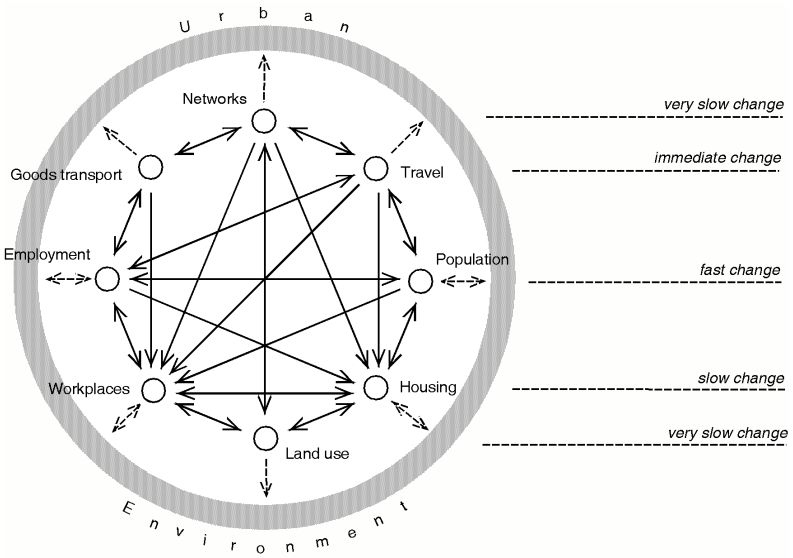


Figure 3.2: A general model of creative strategy [170]

of Coulombel [44]. In this chapter we repeat the most important subjects relating to the scope of this project.

Process

The housing location choice process is often represented as two step structure: decision to move and residential choice[44]. In reality, this involves series of decisions, as showcased in Figure 3.3. Each of these steps are likely to involve their own time constraints (e.g. home search and relocation). For abstraction purposes, these are omitted in our framework, but the integration possibilities should be explicitly considered while building the simulation.

Motives to move

According to Dieleman [50], in short distance residential mobility the primary motive to move is the adjustment of housing consumption (e.g. home size, housing type), which is usually caused by life events (e.g. birth, divorce). However, in our case we assume, that the relocation is caused by the risk of being injured due to unsafe building stock. The households, that are not under risk theoretically should have a threshold of discontent before they choose to move, but the definition of such properties goes beyond the scope of this work. Nonetheless, an interesting example of showing the impact of household and housing characteristics are shown in the work of Kim et al. [90].

3.3.1 Considered criteria

The selection of evaluation criteria can have a considerable effect on the evaluation process: the results can be skewed by including or excluding certain criteria [123]. In addition to that, an increased number of criteria may lead to a more realistic model, but it also leads to increased levels of error through the need to calibrate more parameter [56]. This phenomenon is called Information Paradox. Determining which specific properties matter requires identifying stated and revealed preferences of different population groups [44]. To avoid these whilst depicting choice behavior, we choose to focus on the latter of the two, as this allows us to objectively and quantitatively capture the criteria taken into account (see [53]). The techniques com-

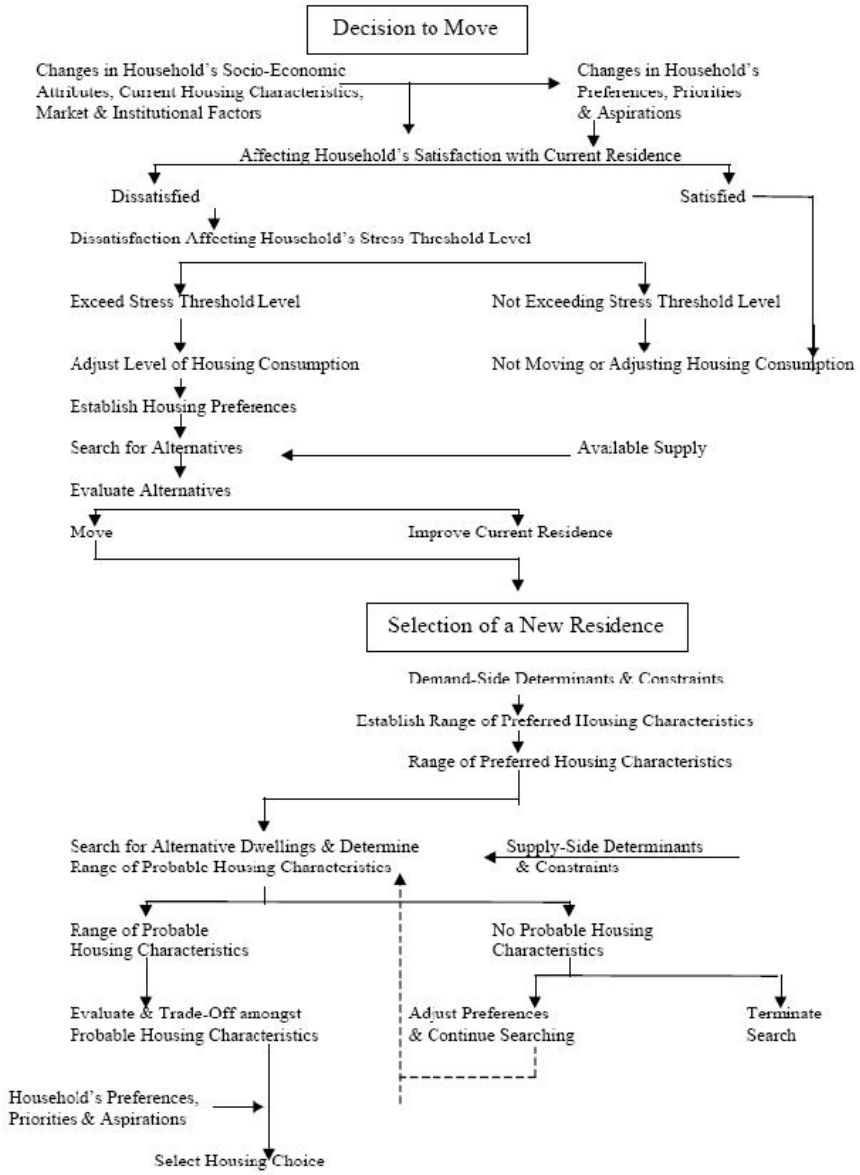


Figure 3.3: Conceptual model of the household's housing decision-making process ([174] in [44])

monly used for this is discrete choice modeling (discussed in later in the chapter) and hedonic analysis (for a great introduction see [143]).

The choice of the house itself can be coupled to two perspectives: location properties (e.g. neighborhood type, location-based accessibility [161]) and dwelling characteristics ([108] in [44]). Interestingly, Coulombel [44], identifies this rather as a "a wall separating the issues of location and dwellings characteristics in academic research" (p.8).

Dwelling and household properties

The dwelling properties, that are commonly considered in residential mobility model include its floor and parcel areas, type, price in relation to household properties like size, employment status, nationality, income, gender, marital status among many. Li [95] (p.32-33) provides an overview of the most popular models (e.g. UrbanSim, ILUTE, PUMA), covering the major features of each of them. A similar overview, but focusing on the dis-aggregate residential mobility models is provided in the paper of Huang et al. [77].

Location-based criteria

Apart from the static house properties (like in [108]), research often considers location-based accessibility as an important measure in residential choice. In this case, models interpret this measure in travel times, thus taking mode of travel and congestion into account, if the measure is part of LUTI (e.g. UrbanSim, MARS). These indicators commonly consider destinations such as parks, shops, jobs, health-care and education (see e.g. [140], a review in [141]). These variables become dynamic, once the households have specific relations to space, such as a fixed workplace or schools locations. In other words, if households are represented individually, the perception of the house location becomes subjective. This is also applicable for static variables, but would require identifying household preferences.

What to represent?

However, referring back to the issue of the limited cognition within the decision making process, we focus on representing only a couple of the afore-mentioned criteria in our model. Just as for visualizations, our goal is to communicate the usability and scalability of the framework, rather than create an accurate representation of the residential mobility processes. Therefore, we identify the need to represent some core static (i.e. housing-bound) and dynamic (i.e. location-bound) variables, which are perceived in the light of limited households heterogeneity.

3.3.2 Rationality

Another thing to consider is the bounded rationality of the entities, that are making the location choice. The theory of bounded rationality suggests, that the actors have limited "information-processing capacities" [145]. These theories postulate, that there is risk and uncertainty, associated with decisions and information on the alternatives can be incomplete (ibid.). Thus models of bounded rationality assume, that decisions are based on a part of the factors and they do not necessarily give objectively optimal results [177]. In this project that comes into play in two ways: that not all attributes are necessarily considered and the reduction of the search scope for the candidate locations.

The first of the two requires observed choice data, such that variables falling in the evaluation can be identified. Provided, that the populations are not homogeneous, different preferences per demographic group should be identified. For instance, families with children are more likely to evaluate quality and distance

to closest schools and kindergarten, whilst that might not be a consideration for households of elderly people.

Regarding the second aspect, in older research (e.g. [108]) it is common to assume, that a household can compare a very limited amount of houses at once (i.e. rarely more than two). However, in our case we identify the advances in information sharing through real-estate websites. More specifically, we assume that households can evaluate all houses falling in their interest range (e.g. they can afford it, sufficient floor area for the number of people) within our research area (i.e. 1/3 of municipality). In simulations dealing with larger areas, researchers have adopted various techniques to deal with this. One of such is distance-based thresholds. For instance, Acheampong [4] introduces a search radius around a household's current location, whilst Rashidi et al. [128] propose a property search model, where properties falling outside the maximum commuting distance are not included in the evaluation for the choice.

3.3.3 Summary

All in all, at this stage, many of the elements of a fully representative residential mobility model go beyond the scope of the project. To name a few, we see life events, economic mechanisms as formation of housing prices, transaction costs, bounded rationality, demographic heterogeneity and change of job/school location going beyond the time or spatial constraints of the project, but strongly supports their integration in future development. Nonetheless, existing research and models allow us to identify the necessary features for the developed framework, as well as the types of variables (location-based, static), that need to be incorporated.

3.4 AGENT-BASED MODELS IN URBAN CONTEXTS

The ever improving computing storage and power has enabled modelers to inspect systems at extremely fine levels of detail, while capturing complex phenomena in a way impossible with aggregate, traditional approaches [76]. One of such methodologies is Agent-Based Model (ABM). They "are computer simulations designed to examine the behavior and interactions of autonomous agents" [144]. Agents in our case would be households, making a decision to move to a location.

This modeling approach has found many applications from supply chain management to finance, archeology and health-care (recent general review [97]). But most importantly, from methodological standpoint, these types of models can be readily applied in complex systems, be it regions or cities [76]. The link between the two is clear– ABMs creates a laboratory for us to observe individual decision-making and interactions, from which emergent, macro scale patterns can emerge (illustrated in[15]). Heppenstal [76] identifies three key advantages these models have over traditional, statistics-based methods:

1. In contrast to ABM, statistical models can represent individual micro-dynamics only if the population is homogeneous or it has coordinated interactions;
2. ABMs allow multiple spatial relationships;
3. Their structure is flexible, allowing incorporation of statistical techniques and mathematical formulations, when the converse is often the opposite.

Nonetheless, most of literature studied dealing with ABM (e.g. [97, 76, 77]) identify similar issues with this modeling technique: the validation, verification and difficulty comparing the results. Heppenstall et al. [76] even states, that calibration and validation in ABMs "remains a dark art". Simultaneously, ABMs can be hard

to communicate due to, among many, complex data structures, modeled entities, or how processes and events are scheduled [69]. As a response to that Grimm et al. [68] proposed a protocol for developing these models in a more standardized way.

3.4.1 Practical considerations

The ODD modeling protocol [68]

The Overview-Design-Details (ODD) protocol, as the name and abbreviation suggests, consists of these three groups. They are meant to lead the modeler in both defining and communicating their model outcomes. Given this framework (Table 3.1), we will present our model in the Chapter 4.

ODD	ODD element	Questions to be answered?
Overview	Purpose	What is the purpose of the model?
	Entities, state variables, and scales	What kind of entities are in the model? By what state variables, or attributes, are these entities characterized? What are the temporal and spatial resolutions and extents of the model?
	Process overview and scheduling	Which entities do what, in what order? When are state variables updated? How is time modeled — as discrete steps or as a continuum over which both continuous processes and discrete events can occur?
Design concepts	Design concepts	There are ten design concepts (see [69]). How have these concepts been taken into accounting the model's design?
Details	Initialization	What is the initial state of the model, i.e. at time t = 0?
	Input data	What input does the model use from external sources such as data files or other models to represent processes that change over time?
	Sub-models	What, in detail, are the sub-models that represent the processes listed in "Process overview and scheduling"? What are the model parameters, their dimensions, and reference values? How were sub-models designed or chosen, tested, and parameterized?

Table 3.1: The seven elements of the ODD protocol [69]

Data synthesizing

The issue of lack of disaggregate data in this project requires to create synthetic datasets. This is because within this project we are unlikely to utilize any micro-census data, but rather several aggregate level datasets. The procedure widely used to deal with this issue is called Iterative Proportional Fitting (IPF), first proposed in [49]. Huynh et al. ([79],p.2), give a brief description of it:

> In conventional population synthesisers, the requirement for these fully joint distributions is that they must preserve not only the correlation between these control variables as observed in the subset of the disaggregated (survey) data associated with the target area, but also the correlation between the marginal distributions of the variables that are specific to that target area. Once the fully joint distributions between all the control variables are constructed, records of individuals in a household

are iteratively drawn from the survey data so that joint distributions of attributes of the resulting synthetic population match as closely as possible the distributions obtained from the IPF process. The households, the residents therein, and their attributes (both at household level and individual level) are stored as part of the resulting synthetic population.

For a more complete picture on the available methods refer to the recent reviews [11, 152]

Verification, calibration and validation

This aspect still remains one of the major challenges within the ABM [165, 142, 99]. Keeping in mind, that calibration is beyond the scope of this project, we want to give a brief overview of it, as one of the sub-questions of the project relate to defining the requirements for calibration data.

According to Paker [119], key to verification is sensitivity analysis. Namely, this entails testing incremental parameter changes against mapped model outcomes in order to ascertain the spatial or temporal limits of a model's applicability and identify possible model errors. This sensitivity is described by the spread of the parameter values, that match specific model results [147]. Once this process is completed, the calibration process is conducted to identify suitable values for the model parameters in order to obtain the best fit with the real world. As we will see in the next section, within the field of Discrete Choice Modeling, this is called model parameter estimation. As last we validate the model for it's representativity of a real-world phenomena. For an introduction in agent based model validation refer to [9]. Moreover, another aspect is very important when talking about modelling– the definition and quantification of uncertainties [142]. In this project we focus on the definitions provided in Table 3.2.

Epistemic uncertainties	- "of or relating to knowledge or knowing" [103] - Resulting from an inadequate understanding. - With time (e.g., additional observations) these uncertainties can be reduced and the true value ascertained.
Aleatory uncertainties	- "depending on an uncertain event or contingency as to both profit and loss" [102] - Due to the intrinsic variability of nature. - Over time, all values will eventually be sampled.

Table 3.2: Uncertainty types (adapted from [60])

Uncertainty due to stochasticity

Many ABMs have some stochastic variables or inputs [100], which means that a single model run gives only a small part of the picture. Namely, it is only a specific realization of the model. To deal with this many ABMs execute many parallel runs with different parameter values. One example of such is UrbanSim, which utilizes random number generators to represent uncertainties relating to choices of household or job location, land-use development [142]. In addition to that, UrbanSim inputs are also estimated by external models, meaning, that the inputs are also not exact. Even though the proposed framework is unlikely to have stochastic elements inside, we do acknowledge the fact, that the inputs are to some extent random. The models explained later in the chapter, Discrete Choice Model(ing) (DCM), are inherently stochastic. More precisely, they describe probabilities of choice, rather than the choices themselves. In addition to that, DCMs include a random error

term, which reflects "idiosyncrasies and particular tastes of each individuals" ([113], p.230).

3.4.2 Residential mobility

Considering these advantages, the ABM has been widely applied in the field of residential location choice modeling. Filatova et al. [61] provides a thorough overview of 51 models. In their work authors identify key elements used within these models:

1. Household agent heterogeneity – these models represent daily activities (travel to work or shopping) and life events, thus their residential choice depends on age, marital status and job location among many.
2. Multiple agent types – most models have more than households as agents, as they simulate location choice of firms and real-estate development;
3. Environmental effects and transport effects (e.g. air pollution, congestion) to support policy and planning analysis

From our perspective, we choose the daily activity representation as an essential feature, but leave the life events and thus household composition as static. Moreover, the functions of the actors beyond households are interpreted twofold. Namely, building stock development is left as user input. Simultaneously, the job mobility is interpreted as static. As last, we see a possibility of integrating the environmental factors in the model by harvesting the values for each of the houses, based on the continuous or zonal representation of phenomenon (e.g. noise, air pollution, urban heat island). Transport simulation would potentially go even further on that, allowing dynamic representation both of the environmental factors and of the earlier mentioned daily activities. Huang et al. [77] also state, that exclusion or limitations of urban market representation, might lead to significantly biased and conflicting results. However, the question that this raises is whether traditional land-market representations can capture the system behavior under disaster risk. Keeping in mind, that this goes into the field of urban economics, we are conscious about land market representation integration in the future, but do not attempt to integrate it ourselves.

3.5 DISCRETE CHOICE MODELING

The field of discrete choice modeling has been incepted in the 1970s [101]. Since then has been applied throughout numerous contexts, such as health care, consumer choice and transportation [40]. At the core of these studies are quantitative analyses and predictions of choice behavior. Having data on observed choice behavior and the properties of the available options, one can derive the underlying preferences of individuals. More specifically, identify the weights given to each of the property types during the choice procedure. In this section we will discuss two theories in DCM, for full overview and application examples refer to [130].

3.5.1 Utility-based models

Majority of discrete choice models are based on the principle of utility maximization [40]. These types of rules evaluate the utility or benefit of each of the options and choose the one with the highest value. The models associated with this are called Random Utility Maximization (RUM) models, where the 'randomness' is represented by the error term added to the evaluation rule. This term is introduced to account for the uncertainty arising from incomplete information on the decision

process and it elements. Ortuzar and Willumsen [113] state, that the random term allows the modelers to explain the two 'irrationalities' inevitable in human behavior. First, that two attribute-wise identical individuals may choose differently when presented with the same options. Secondly, that some individuals may not choose the best option from the model perspective.

$$U_i = V_i + \epsilon_i = \sum_m \beta_{mj} \cdot x_{im} + \epsilon_i \tag{3.1}$$

Where:

U_i	random/ total utility associated with a considered alternative i
V_i	observed utility associated with i
ϵ_i	unobserved utility associated with i with assumed mean of 0,
β_{mj}	constant for all individuals in the homogeneous set, but may vary across alternatives
x_{im}	value associated with attribute x_m for the considered alternative i

This evaluation can then be used to determine the probability of an option being chosen. Commonly this is done by utilizing Multinomial Logit (MNL) model, which is "the simplest and the most popular discrete choice model"([113], p. 232). It can be generated assuming, that the random residuals (ϵ_i) are independent and identically Extreme Value (EV) type 1 or IID Gumbell distributed (variance of $\pi/\sqrt{6}$)

$$P_i = \frac{exp(\theta V_i)}{\sum_j exp(\theta V_j)} \tag{3.2}$$

where θ indicates the sensitivity for differences (in practice normalized to 1, as it is simultaneously estimated with β(discussed in [113], p. 232 [1])) and is related to the EV1 deviation by:

$$\theta = \frac{\pi}{\sigma\sqrt{6}} \tag{3.3}$$

Since their introduction in the 80s [101] these models have been extensively applied in fields of consumer choice, transport modeling and education, to mention a few (see [22]). However, with this kind of approach it is assumed that humans behave rationally. The contrary is evident, in a sense that people have subjective preferences, limited cognitive abilities and limited time to decide (topic widely discussed in [146]). The aspect of this problem, in a sense of regret aversion, is addressed in the next section.

3.5.2 Regret-based models

Regret is a phenomenon, that arises when a decision maker is faced with a situation, where one or more non-chosen alternatives perform better than the chosen option on a single or more attributes. The basic notion of RRM models is that regret plays an important role in choice behavior [88, 176, 43]. More specifically, the decision makers choose an option, that provides them with smallest regret. Most importantly, this technique allows one to let go of the underlying differences between the measures of housing properties and accessibility measures [44].

Regardless of it's relatively short history (introduced in 2008), RRM has seen many applications, ranging from departure time and route choices, to on-line dating [37]. According to Jang et al. [80], the RRM models, introduced by Chorus et al. [39], were an extension to the seminal work of Bell [20] (management sciences),

1 In this reference they notate the preference weight as β and the θ as a parameter associated with the individuals in the homogeneous set

Loomes and Sugden [96] (micro-economics), and Quiggin [127] (general) [2]. The improvements allowed to move from binary to multinomial logit models and consider multiple criteria choices, making the models more applicable in practice.

Procedure

The RRM procedure incorporates evaluating alternatives in context of the choice set Figure 3.4. First these comparisons are done on attribute level and then on aggregate level. In RUM models, the procedure is similar, but the alternatives are only compared on the aggregate level (see [36]).

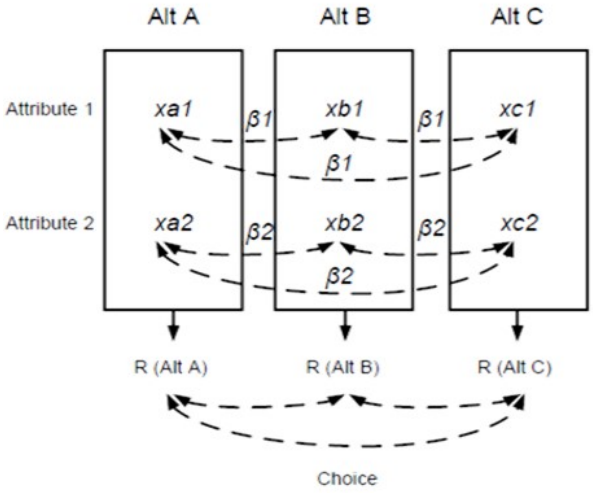


Figure 3.4: Random Regret Minimization decision process principle (solid arrows represent summations, dashed arrows represent comparisons) [36]

Mathematically it is expressed as Equation 3.4, introduced in [38]. It allows quantifying the differences between a set of options on several attributes, which are not necessarily of the same dimensionality (i.e. think of dimensional analysis, adding pears and apples). This is because, we are creating not a linear-addition of the Δ_{xm}, but rather a log-sum. Thus the formula below can also be expressed in the form of Equation 3.5. Additionally, the formula can both deal with minimization and maximization simultaneously. All it requires is the change of the sign of the β parameter– for minimization $\beta < 0$.

$$RR_i = \sum_{j \neq i} \sum_{m} ln(1 + exp(\beta_m [x_{jm} - x_{im}])) + \epsilon_i \tag{3.4}$$

Where:

RR_i	random regret associated with alternative i
β_m	estimable parameter associated with the attribute type m
x_{im}	value associated with attribute x_m for the considered alternative i
x_{jm}	value associated with attribute x_m for the considered alternative j
ϵ_i	unobserved regret associated with i

$$RR_i = \sum_{j \neq i} ln(\prod_{m} (1 + exp(\beta_m [x_{jm} - x_{im}]))) + \epsilon_i \tag{3.5}$$

2 According to the researchers in the original research group it is not an extension, but a 'new' model on regret [38]

Having the 'observed' part of regret allows modelers determine which alternative is best (i.e. rank on smallest regret). However, just like RUM, the RRM formulation also includes an 'unobserved' part of regret ϵ_i. Given that it has a independent and identically distributed Gumbell distribution, MNL can be applied, but with slightly different formulation, as we are trying to minimize the values and the preference parameter β_m is already incorporated in the regret function:

$$P_i = \frac{exp(-R_i)}{\sum_j exp(-R_j)} \tag{3.6}$$

RUM versus RRM

The biggest difference between the two modeling approaches is their approach to compensatory behavior or how much rejoice plays a role in compensating the experienced regret. To illustrate that in spatial terms we investigate a situation of a household of two individuals selecting a location for their house. The only two aspects these individuals consider, are their travel distance to their work locations, which are in two separate cities A and B (Figure 3.5). This implies, that the household will try to optimize 2 criteria, which are the travel times to A and B. We assume, that both of these attributes have the same preference weights. If we would take RUM approach, the location of the house would not matter, as long as it was between the two cities. This is because the total travel distance or total dis-utility would remain the same. However, when we take the linear-additive RRM, the optimal location will always be in the middle between the two cities. This is so, because the reSgret function is not linear, but logarithmic (Figure 3.6). As we will see later in the chapter, RRM can capture both compensatory and semi-compensatory behavior, due to the controls built in the model.

Figure 3.5: Example of the optimal location choice given the travel distance on a network of a single edge

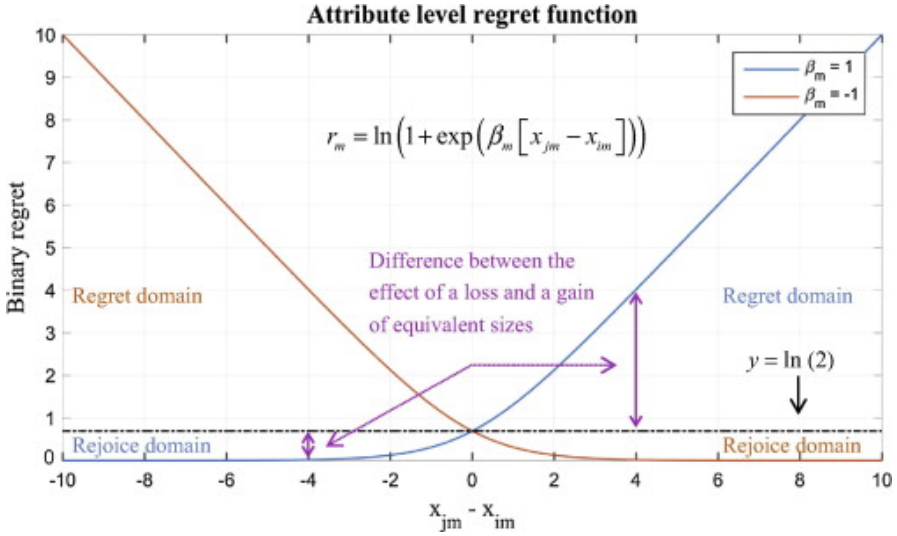


Figure 3.6: Attribute level regret function [158]

3.5.3 Limitations and alternatives

The classical RRM model imposes a behavior, that varies between datasets and attributes. This is caused by their scale invariance [158], as the taste parameter β not only reflects the importance of a criteria in the decision, but also the shape of the attribute level regret function. What it means, is that even while having the same dataset, but with attributes evenly scaled, it will return completely different outcomes. In order to capture the attribute importance and the degree of regret separately, Van Cranenbergh et al. [158] propose μRRM model.

μRRM

The μRRM model generalizes the Classical RRM model by allowing the variance of the error term to be estimated [158]. More precisely, in the μRRM model the scale parameter μ is added (Equation 3.7). Noteworthy, is that we are using the formula found on the website of the authors (advancedrrmmodels.com) and communication with them, rather than the paper, introducing the theory [158]. There the formula does not have the first μ parameter.

The parameter is linked to the error variance and allows one to estimate the shape of the attribute level regret function (Figure 3.7). With this change one can indicate the levels of profundity of regret. This term refers to the extent the model imposes regret minimization behavior. Thus the larger μ values, the milder the regret minimizing behavior. Thus if the parameter is arbitrarily large, it will exhibit the linear-additive RUM behavior. The other boundary case, when $\mu \to 0$, shows the Pure-RRM behavior, which shows the strongest profundity of regret within the framework (see [158]). When $\mu = 1$, the model 'collapses' to the classical RRM model [38]. However, even though μ is added, it does not remove the influence that the (scale of the) units have to the decision rule. More specifically, as in the classical RRM model, the βs and the μs will have to be estimated for each dataset and model application.

$$RR_i = \sum_{j \neq i} \sum_{m} \mu_m \cdot ln(1 + exp(\frac{\beta_m}{\mu_m}[x_{jm} - x_{im}])) + \epsilon_i \tag{3.7}$$

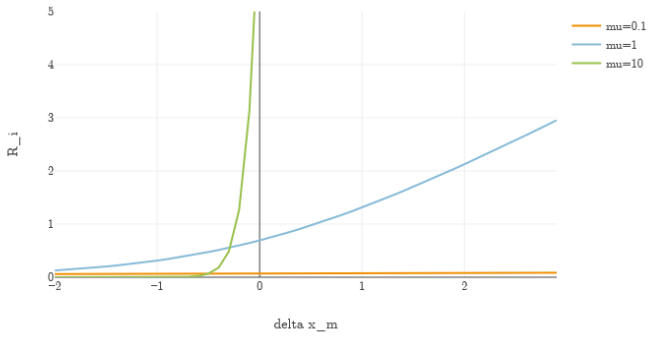


Figure 3.7: The impact of μ on the binary variable level regret function, $R^{\mu}_{im} = \mu \cdot ln(1 + exp(\frac{1}{\mu}[\Delta x_m]))$, where $\Delta x_m \in [-2, 3]$

RRM^{PLW}

Another take on the RRM is provided by Jang et al. [80]. The authors propose interpreting regret from the perspective of Weber's law: "individuals perceive change in a stimulus as a constant ratio of the original stimulus" (p. 5 of 21). The Paired Logarithmic Weber's law (PLW) formula takes the form of the following:

$$RR_i^{PLW} = \sum_{j \neq i} \sum_m ln(1 + exp(\beta_m^{PLW}[\frac{x_{jm} - x_{im}}{x_{jm}}])) + \epsilon_i \tag{3.8}$$

The benefit of using this formula is that it makes the calculation dimensionless without the β estimation and solves the scale invariance issue. I.e. the larger the attribute size, the smaller the regret with respect to the same attribute-differences. However, this specific application does not work on datasets with many attribute values equal to 0. A workaround for that would be to uniformly shift the whole attribute value set, so that none of the values are 0. However, we consider a different approach with very similar implications and no need to manipulate the input data.

RRM^{σ}

In this research we propose an alternative that allows us to deal with both the scale invariance and the dimensionality issues. Instead of dividing by the original value, we propose scaling the attribute sets by their standard deviation. This measure captures the spread of the attribute set. The formula then takes the following form:

$$RR_i^{\sigma} = \sum_{j \neq i} \sum_m ln(1 + exp(\beta_m^{\sigma}[\frac{x_{jm} - x_{im}}{\sigma_{im}}])) + \epsilon_i \tag{3.9}$$

The downside of this method is that calculating the standard deviation of the set is more computationally intensive. The upside of using this formula compared to the RRM^{PLW} model, is that the standard deviation of a set of alternatives is rarely zero and, if it is, then the attribute level regret is identical. This allows us to bring in easy controls in the scripts whilst utilizing this function. It also retains the benefit that it makes the calculation dimensionless without β estimation. Meaning, that the β values can become a more intuitive measure of the importance of an attribute, as it also can become dimensionless. Potentially, it could also include the μ, which could further increase the controls of the choice behavior (i.e. the profundity of regret). However, in our research scope that does not play a role, as we have no observed behavior data and assume that both the βs and the μs are equal to 1.

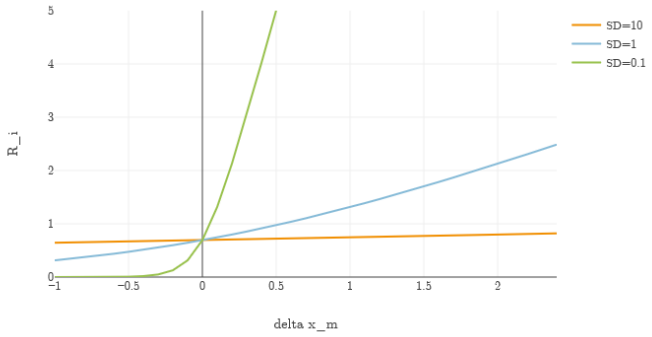


Figure 3.8: The impact of σ (standard deviation of m) on the binary variable level regret function shape, $R_{im} = ln(1 + exp(\frac{[\Delta x_m]}{\sigma_i m}))$, where $\Delta x_m \in [-1, 2.5]$

Comparisons

We examine the unit scale implications of the two alternative models and the classical RRM in a 'toy' problem (Figure 3.9). The example choice set has 3 abstract options/alternatives to choose from, each of them having two properties: distance and price. However, the price is indicated in two different ways: cents (blue) and euros (orange). This allows us to form 2 separate datasets, which essentially have the same information.

We first inspect the regrets experienced for one of the alternatives. The classical RRM model (No Scaling) shows significant differences in regret, when comparing the inputs with different units, regardless that they contain the same information. This was not the case when scaling the choice dataset with either the reference (x_i, Weber's Law) or the standard deviation of the attribute sets (STD Scaling), as they are unitless.

When inspecting the experienced regret on all the choices, we see consistency between the scaled datasets, as they choose the same alternative (i.e. alternative 1). But we observe that in the classical RRM model the impact on the scale of regret are not relative. Specifically, the scale of units also have influence which of the options is chosen: if the price was indicated in cents the regret would be lowest for option 1, but if in euros it would be the lowest for option 2.

On the issue of unit scale

In line with literature [158], this example showcased the importance of the estimated β parameter in scaling the regret. However, this also means, that β is not transferable through datasets and can only be identified after the estimation. This is a significant issue when utilizing the model in agent-based residential mobility, where the choice sets (i.e. numbers of attributes and their scales) differ per agent and time unit due to the different perception of the building stock. In this case, we would expect similar choice behavior observed between demographic groups, which would need to be captured by the β in a sense that is comparable through different choice sets. Namely, how important is the attribute in the formulation of regret and consequently making the choice for a specific demographic group.

In the classical RRM the estimated β values will be different for each and every dataset. This is due to its connection to the scale of the units of the dataset. This is completely acceptable in the cases where the choice set remains constant and even allows identifying the effects of changing the variables. Namely, it allows answering the question of how the regret will change, if an attribute is increased by x amount of units. This is particularly interesting as an application in transport modeling, as

this would allow identifying the choice behavior changes in relation to changing pricing policies.

However, inconsistent with the statements of the researchers [158], a similar situation can be observed in μRRM, where this is captured in two estimable parameters of β and μ. These issues become clear, when inspecting the theory from the dimensionality perspective (i.e. units). Keeping in mind, that we are provided conflicting information on the theory, we inspect both formulas provided (i.e. from [158] and from author's website and communication). For the illustration, we drop the error terms, as they are of the same unit as regret, as they are associated with the 'unobserved' part of regret for an alternative i. The first one is the formula provided in the communication:

$$R_i = \sum_{j \neq i} \sum_{m} \mu_m \cdot ln(1 + exp(\frac{\beta_m}{\mu_m}[x_{jm} - x_{im}])) \tag{3.10}$$

One of the parameters needs to remove the units associated with the variable. In classical RRM, that is done by β. In this case, assuming, that μ has units would be incorrect– the parameter placing outside of logarithm suggests, that these units would be retained on attribute levels. This implies, that the sums happen on values of different dimensionality (e.g. number of windows and distance to a closest shop), which would be incorrect in a mathematical sense. This suggests, that μ is actually unitless and β still captures the unit scales. This suggests, that even though the μ allows another level of freedom for defining the shape of the regret function, the formula is by no means scale invariant, as the definition of β is the same as in Equation 3.4.

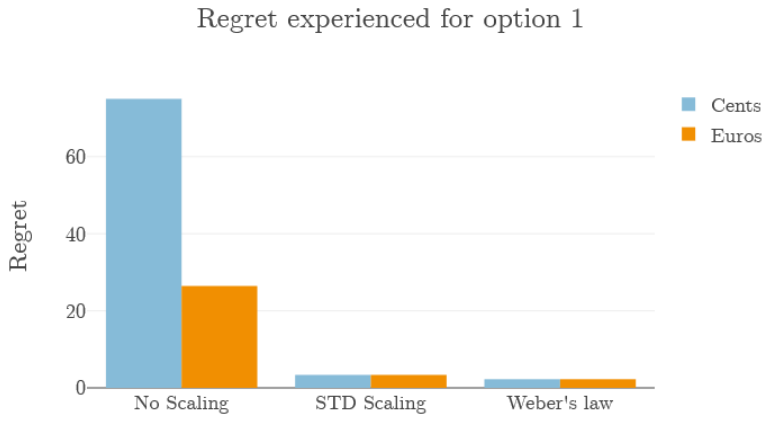
Looking at the μRRM definition in the paper [158] (Equation 3.11), this is potentially not the case.

$$R_i = \sum_{j \neq i} \sum_{m} ln(1 + exp(\frac{\beta_m}{\mu_m}[x_{jm} - x_{im}])) + \epsilon_i \tag{3.11}$$

The authors also claim, that "by dividing by the scale, the estimated taste parameters are adjusted for the scale of the μRRM model" (p. 97). The authors further clarify, that this allows to straightforwardly compare the taste parameter estimates across different models. These statements indicate, that the β becomes a unitless attribute preference indicator, as the μ is the one that captures the scale of the units. However, this is contradicted by the definition of the term, where it is linked to the variance of the error term ($\sigma^2_{\epsilon_i}$): $\mu = \sqrt{\sigma^2_{\epsilon_i}/(\pi^2/6)}$ (p. 96). According to this definition, μ will have the same units as the error term, i.e. the 'unobserved' part of the regret. Implying, that in the formula, the β is still the parameter removing the units within the logarithm. This again showcases, that this model definition is not unit scale invariant and that the βs could not be compared across different models, with varying attribute datasets.

3.5.4 Other considerations

Regardless of all the refinements RRM brings in for the behavioral representation, its computing complexity is inferior to RUM. In other words, while RUM would scale linearly with the choice set, the RRM scales quadratically ($O(n^2)$) (apart from P-RRM [158]). This brings a challenge on the application of the theory on large spatial regions. The proposed σRRM requires even more computations, as for each evaluation iteration one needs to compute the standard deviation of the set, which would suggest using RRM^{PLW}. However, the latter approach also does not allow for easy integration within the scope of residential mobility under disaster risk, as



Alternative	Distance (miles)	Price (cent)	Price (eur)
1	50	100	1
2	75	50	0.5
3	20	150	1.5

Figure 3.9: Demonstration of the scaling issue for the classical RRM, RRM$^{\sigma}$ and RRM^{PLW}

many attribute choice sets are likely to contain zeros. Such as, when the building is not under any risk. Utilizing standard deviation, even if there are no differences between observed variables can be easily accounted for and thus replaced by the attribute value itself or even by automatically skipping the evaluation and giving the attribute regret a constant value of ln(2). Therefore, we select this theory for our framework presented in the next chapter.

3.6 SUMMARY

In this section we discussed the context of this research, relating to a wide range of scientific fields. We started by showing that models play an important role in public planning and policy making process and that there are many other tools apart from simulations to support it. Nonetheless, when talking about cities as complex systems, models can play an invaluable role as a controlled laboratory for experimentation. This allows public sector agencies test their proposals in a controlled manner, showing the possible pitfalls and leading to a more structured dialog between involved parties.

Next, we gave a brief overview of cities as collections of systems, which are interrelated to each other– residential mobility being one of the sub- or partial systems. With this in mind, we continued to the outline of the core concepts and definitions in the field. Additionally, we strove to relate each of them to the scope of this research and give indications what future improvements should be taken into account while building the framework.

After this we continued by explaining the reasoning for choosing a disaggregate modeling approach utilizing agents. We presented the core definitions and pitfalls of the modeling technique through the prism of the ODD protocol. As last, we went in depth of discrete choice modeling, which is an essential feature to replicate agent behavior in the perspective of residential mobility. We briefly discussed the predominant utility maximization modeling approach and gave an alternative, based on the notion of regret. Given the already existent regret-based models, we analyzed the difficulties of application given our problem, primarily relating to the

unit scale-invariance. As an alternative, we proposed a novel approach dealing with the unit scale issue and allowing for showcasing its application within the field of agent-based residential mobility.

4 METHODOLOGY

> In anything at all, perfection is finally attained not when there is no longer anything to add, but when there is no longer anything to take away
>
> *Antoine de Saint- Exupéry* [48]

The framework we are presenting in this thesis can be seen as a process consisting of 4 steps (Figure 4.1). In the first step (Section 4.2), we define the model input. In the second (Section 4.3)– perform the simulation of household relocation . The third step (Section 4.4) allows for interventions or input for the model, enabling us to provide alternative inputs for the simulation. Finally (Section 4.3.5), we collect all of the outputs and provide them in the form of an series of interactive visualizations.

This section will be structured in a similar manner. We will start by giving an overview of the key features of the framework and then proceed by discussing the data-related topics: data structure and processing approach. Next, we introduce the model definition and go into more detail relating to specific methods or functions in the code. After this we explain the experiment setup: the spatial extents studied and the interventions chosen for showcasing of the model. The chapter finalizes with description of the Verification & Validation (VV) methods, which zooms in on identifying model uncertainties.

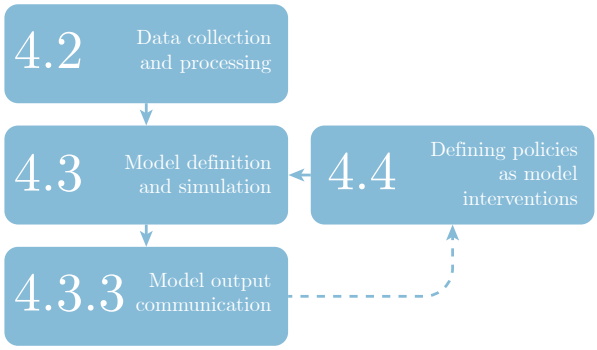


Figure 4.1: Framework overview with chapter numbers (lines show procedural flow, with dashed line representing a thought process; people getting ideas based on the visualizations)

4.1 OVERVIEW

The proposed model framework is intended to enhance the strategic planning capabilities of Regional Planning Organizations and other state and local agencies. However, within this research we are not creating a predictive model, but rather showcase the possible applications for testing hypothetical or 'What-if' scenarios for their proposed policies and spatial interventions. By this we imply, that the framework

would enable decision makers to identify possible pitfalls of planned interventions in the housing market (zoning) and financial policies to support housing-stock upgrading. To do so, we introduce an agent-based residential mobility model, which has the following features (based on the Overview-Design-Details (ODD) protocol):

1. Overview
    a) The model simulates one of the key choice makers' (households') choices, underlining bottlenecks in the current configurations of the urban development; Households are thus the agents in the model
    b) The agents of the system are households, which can relocate to discrete locations (parcels and houses), connected by a network; the collection of these elements is a system
    c) The time representation of the model is an abstract time unit called step; during each step households evaluate viable (i.e. affordable, sufficient in size) building options, which of the best one is bid upon; the bid is always won by a household with the largest capital; the households then move to a new location, vacating their old building for evaluation for the next step; system attributes are collected after each successful bid and the process repeats until convergence (i.e. no more relocations) is achieved.
2. Design concepts
    a) The model is based on random regret theory and uses option ranking for agent residential choice to relocate and where to relocate.
    b) The model addresses re-development/upgrading (intrinsic, built in the model) and new developments (extrinsic, user input) of the building stock.
3. Details
    a) On initialization the households are moved to semi-random starting locations (based on aggregate statistical counts); a run without change to this input is called a base run; making a change to the input data constitutes an intervention or a scenario;
    b) The input data comes from public datasets (i.e. aggregate statistical data, building stock properties, network geometries) and private host company datasets (i.e. risk and building functions (result of integration of public and commercial datasets))
    c) The simulation is performed on sets of households, which are synthesized based on aggregate statistical data; the household sets are the same for all interventions and their properties change only, if specifically indicated in the intervention;
    d) With the abstract model time representation, we assume, that the demographic situation, housing and job markets are static.
    e) The user interface focuses on interactive evaluation of pre-defined 'What-if' scenarios
    f) The model and user interface are written in Python, using object-oriented programming to maximize software flexibility; the user interface, a dashboard is browser-based.
    g) Data processed, synthesized and produced for and by the model are written as HDF5 files (open format) for external use.

Additionally, within the scope of this thesis we select a minimal number of required optimization criteria to showcase the functionality of the model. In this

work we refer to criteria when talking about system objectives, i.e. criteria to be optimized. The term of attributes refers to the data itself, i.e. the building value. In relation to each other, an attribute can be a criteria, but it is not always the case the other way around. Namely, a criteria could be expressed in a function form, integrating one or more attributes.

4.2 DATA COLLECTION & PROCESSING

4.2.1 Data structure

To make the work easy to expand and edit, the model is implemented in an object-oriented manner (Figure 4.2). That implies, that we are structuring our code in smaller modules: classes. Each of them have specific attributes, called fields, and methods, or functions. Our code includes 4 major classes:

- Buildings – stores all variables relating to buildings, as well as their relations to parcels (abstract class) and thus the network.
- Households – stores all population and household related variables, methods for population synthesizing.
- Model – iteratively performs the buildings evaluation from each household's perspective and relocates households until a termination criteria are met.
- Network – describes the geometry of the network and stores the distance matrix to every point of the network.

Other classes in Figure 4.2 are abstract. More specifically, they exist as blueprints for other classes. For instance, *Household* abstract class would define specific properties of a single household, which do not work as single entities in code, but form the *Households* class (collection of households). Noteworthy are the classes *Changelog* and *Visualization*. The first of the two is defined in the *Model* class and represents the storage of the changes that occur during the *Model* class execution. The second– *Visualization* class– is a series of methods used to create the interface to explore and analyze the model outcomes (i.e. *Changelog*) for different interventions and the base run.

4.2.2 Data preparation

The first stage in the pipeline relates to selecting and processing the data. We select the datasets based on literature review, visual and rounding error inspection (covered in Chapter 5). Here we provide a flowchart per class with the final selection of datasets. The color coding in them correspond to the following:

- *blue* shows the procedures carried out
- *gray* points to external data sources (overview in Table 5.1);
- *green*– relations to the other classes;
- *red* marks the resulting file.

Network class

The space in this model is represented as an undirected network of streets, which connect all the objects in space. This data is retrieved by using the pipeline identified in Figure 4.3.

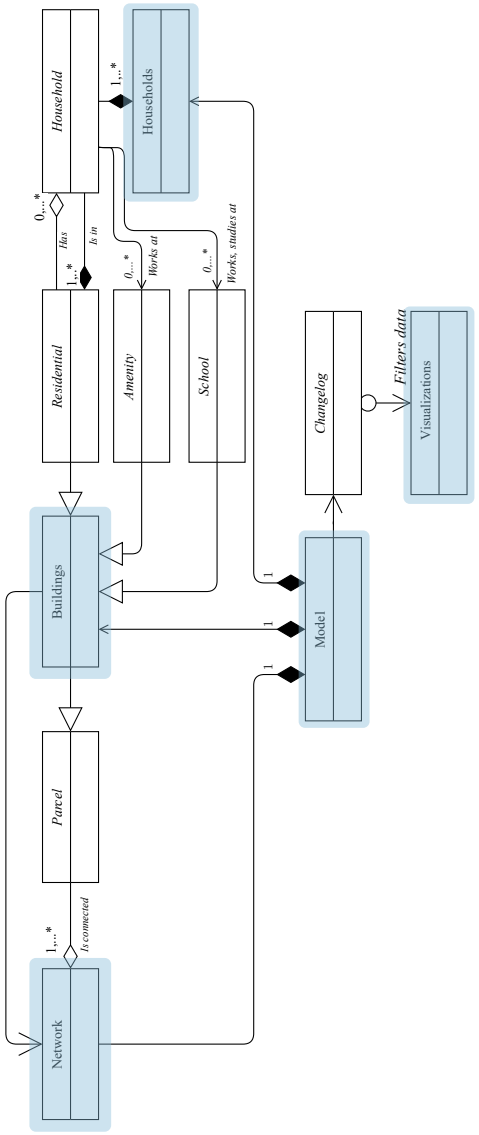


Figure 4.2: Simplified UML diagram of the project, hard-coded classes are marked in blue (full version in appendix A)

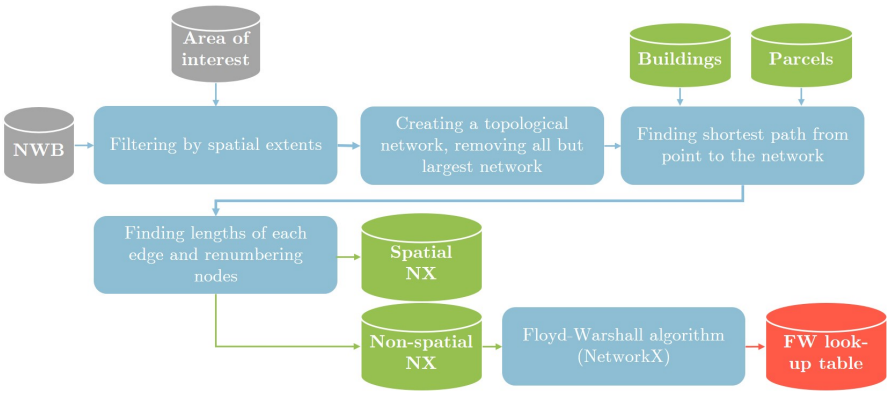


Figure 4.3: Network class processing

The first step in our approach is to (1) filter the line geometries falling outside the case-study extents. This implies, that some of the lines would form disconnected sections. To circumvent that, we first (2) construct a topological network representation, where each intersection or point becomes a *vertex* and lines between those vertexes become *edges*. After performing built-in connectedness tests we discard all but the biggest network. This is acceptable, as we are working in administrative unit zones and the subsets of disconnected vertexes are relatively small (processing municipal scale: streets with no objects connecting to them) or non-existent ("buurt" and "wijk" levels).

After this, we can (3) connect the center points of building and parcel geometries to the street network. This is done by finding the center of mass of each geometry at hand and finding the closest point on the edge on the network. These edges and vertexes are then added to the network.

Next, we (4) renumber each edge and vertex, to ensure consistency and calculate the length of each edge. The outcome of this step is saved in two files: spatial, which is used in visualizations; non-spatial, which is used to perform calculations of the model. This is done to speed up the processing of the Floyd-Warshall (FW) matrix construction algorithm (see Algorithm A.1), as the spatial data is not needed for it. The output of this step, also used in the *Model* class, is the Floyd-Warshall shortest path matrix or distance look-up table.

Parcels class

Parcel, or plot of land, processing is carried out in order to identify all empty parcels with the possibility to build residential buildings in them. Following the graph in Figure 4.4, we begin with (1) filtering by spatial overlaps. We select only parcels in the area of interest, that overlap with spatial plan zones, allowing for residential buildings to be built. In this step we also filter out all plots of land that have buildings on them. This allows us to create the bindings between parcels and buildings datasets, adding a key relations between them.

Following this, parcels are (2) intersected with the road network, so that plots designated for roads are removed. This last operation also removes parcels, which were recently destined for new construction, but are not yet subdivided. In the case study area we observed at least one occurrence of this, which is later used to manually create a new neighborhood intervention scenario, described in Section 4.4.

As the last cleaning operation, we (3) check for parcel geometry circularity: if an object is close to a square, the value will be close to 1. This filter is necessary, as the road and parcel geometries do not always align. This step allows us to identify any free parcels, which are (4) connected to the network as described above.

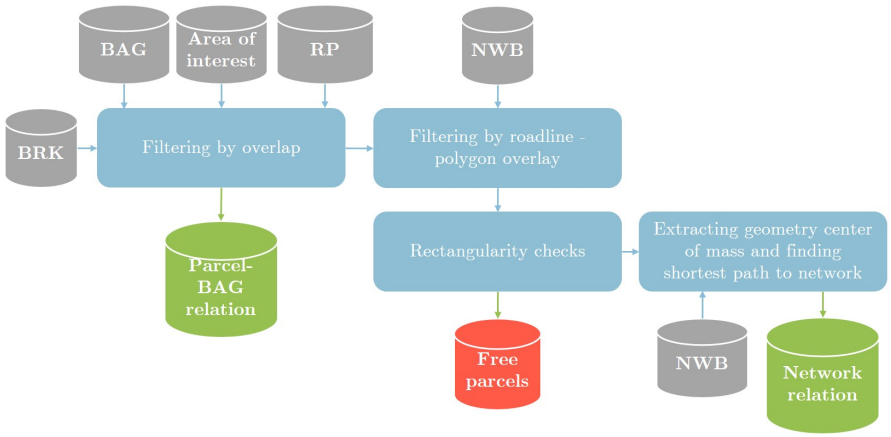


Figure 4.4: Parcel class processing

Buildings class

In the presented framework, we interpret buildings as collections of living spaces or flats, corresponding to the smallest housing unit (Verblijfsobject (VBO)) in the key registries in the Netherlands (see Figure 5.1.2). The processing of this data starts with (1) spatial filtering. We first select only the geometries falling within our area of interest. Then we create relations with the parcel dataset by performing a spatial join.

Next, we (2) connect the building dataset to the host company databases, giving us information on building functions, inferred structural types and risk. The latter data is incomplete and only allows us to tag the buildings with the highest collapse risk. For showcasing purposes, we extrapolate on the available information to quantify and expand the risk tags. In line with the methods used to define the hazard in the region (explained in the [1]), we assume that the most important characteristic influencing building's seismic response are the construction materials and structural systems. We abstract and simplify the risk estimation by assuming that these properties are identical for the whole research area. Our interpretation makes the risk a variable solely dependent on the building's (most probable) structural system (explained in [41]). This implies, that the original highest risk batch of buildings remain at largest risk (unit-less value of 1), whilst the rest of the building stock having the same structural types are given values of 0.5. Buildings not falling in the latter two categories are assumed to be under no risk. To circumvent this data impacting the real-estate market, the proportions of the tags remain accurate, but the risk attribution is randomized. In the same step, the upgrading costs per m^2 are generated for each building under risk. The values are randomly generated using indications of expert engineers in the field: we assumed a standard distribution with a mean of 1500 eur and standard deviation of 500 eur.

After the risk attribution, we (3) connect the geometries to the network, which allows us to (4) find distances to 3 closest amenities (i.e.) to each of the residential buildings and aggregate them. At the current implementation, where facility locations do not change, this becomes a pre-computed, static variable.

As last, we (5) add the workplace location data. At this stage of the project the data was only available on aggregate levels (more in Section 5.1.2). Therefore, we randomly attribute them to the buildings, with identified non-residential functions. This is done a single time, as workplace number per building is a piece of data, which is possible to acquire. Just like amenities data, this variable remains static (i.e. people do not change their workplaces).

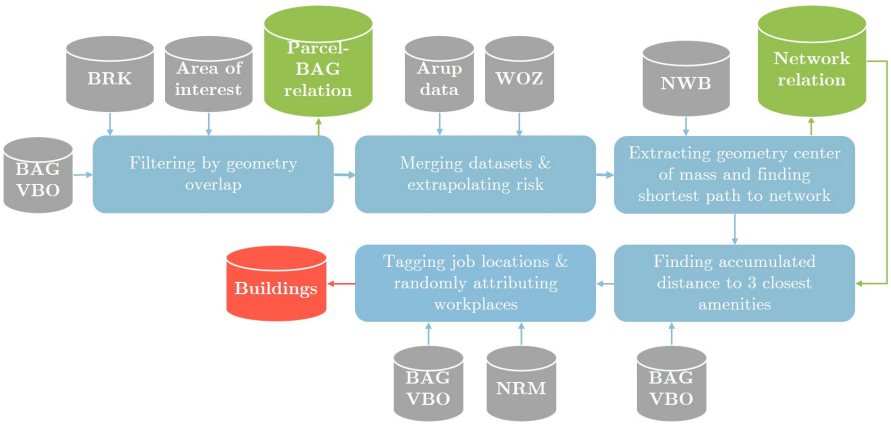


Figure 4.5: Buildings class processing

Households class

The goal of this part of the process is to create synthesized sets of households, which still correspond to the global statistics in the studied areas. This type of approach allows us to create semi-realistic datasets without accessing private data (Figure 4.6).

Just like for all other classes, we first select the zones within the studied subset and (1) extract the aggregate level statistics on demographics (population counts for two groups: adults and children). In the same step, we randomly distribute these people through households, based on the statistics on household types (i.e. counts of single-person, with children and other households).

Due to incomplete building data (see Figure 5.1.2), we reduce the scope of the model by removing the buildings with missing values. This consequently requires to (2) reduce the number of households to ensure realistic vacancy rates.

Once this is done, we proceed with (3) randomly attributing people to houses. Since the demographic data only identifies income segments (i.e. low, medium, high and average income per zone), we assume that the household capital should be in similar ranges as the real-estate they own. As (4) capital we take 110% of the building price, so that households had the capacity to upgrade and move to different locations. This is a simplified approach and would otherwise require complex economic models (e.g. like in UrbanSim) or more refined data input.

As last, (5) from three closest schools (the capacity and types are unknown at this stage) we randomly select one for each child. In the same loop we also find job locations for each of the households, which is based on a job distributions, based on doubly constrained or gravity model with equal travel costs (see [113], p. 182, see Figure 5.1.4). This allows us to synthesize data on aggregate level job mobility. More specifically, the proportions of people working in one of the zones, given their residence location.

To sum up, in this subsection we describe a pipeline for creating synthetic households, which still correspond to global system statistics. The household heterogeneity is encapsulated in their relations to the network (where people work or study), age group (i.e. working, pensioneer, child) and income group. This procedure is repeated multiple times (number dependent on the researched subset) to create many alternative sets, which are then run in parallel to create a spectrum of outcomes for the model runs.

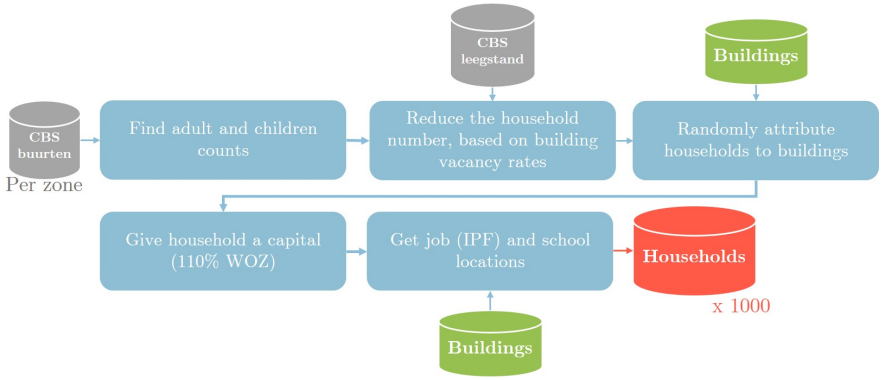


Figure 4.6: Synthesizing of households

4.3 THE MODEL DEFINITION

After collecting the data for all input classes (i.e. households, houses and network), we proceed to the simulation. As seen in the overview Figure 4.7, the preprocessing step (marked in blue) is also part of the framework. In other words, if the files for the preprocessed data are not available on a machine the code is running on, it will be generated. The *model* class (marked in orange) is initiated by collecting all the input data and altering it to represent an intervention (i.e. a scenario). If this data is not provided, such a model instance is called a *base run*. We use it to compare the intervention performance and see if any impact (positive or negative) is achieved.

The simulation starts by executing the **run** function, which contains the **step** method and the information of the termination criteria (explained in Section 4.3.1). The **step** (Section 4.3.2) method is function a level lower, representing the discrete time unit of the model. In that function we go through all the households to run the lowest level **building evaluation** function (Section 4.3.3). Each time the **step** function finalizes, we collect the changes to the system in the *changelog* class.

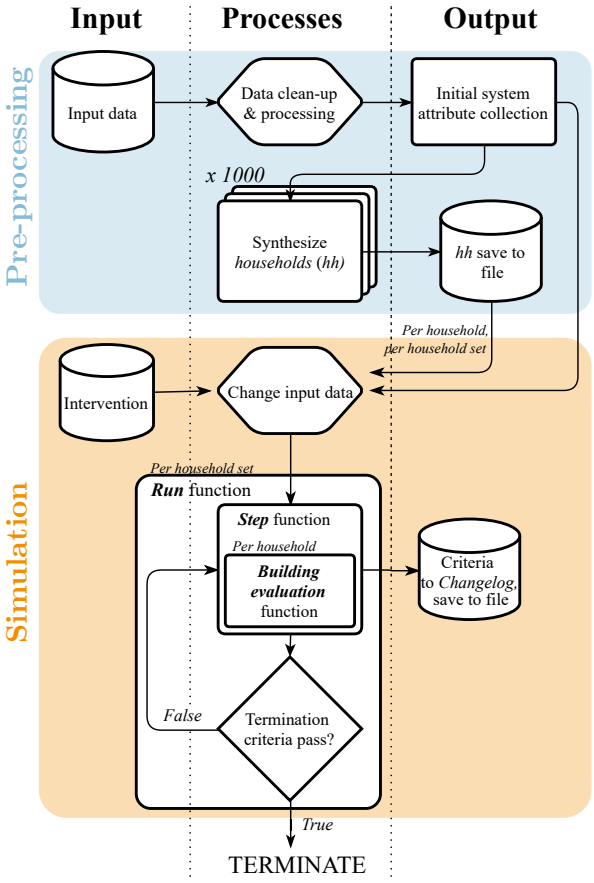


Figure 4.7: The model code flowchart with 3 major functions in the code, in blue– data preprocessing, orange– simulation

4.3.1 Run function

This method is the highest level function in the *model* class. It takes the instantiated object and controls the simulation. More specifically, it initiates the step function and terminates the model, when termination criteria are met. In this project we have used 3 different termination rules: (1) step limit; (2) change threshold with a moving average; (3) convergence (there is a constant number of changes for x amount of steps). Each of them can be inspected in the Algorithm 4.1, with comments as markings for each.

4.3.2 Step function

The step method (Algorithm 4.2) corresponds to the model representation of discrete time. In this function we (1) scan and evaluate the available building stock from the perspective of each household, (2) select their candidate location and (3)perform bidding.

The first of the three utilizes the adapted version of RRM (Equation 3.9) and is explained further in the section. After executing the evaluation step, we can either perform option ranking (i.e. select house with the smallest regret) or utilize MNL. This formula is used to calculate the choice probabilities P, which are then used to select one of the houses from the alternative subset. The last of the three steps is simplified to the extents, that the household with the most capital always wins. Once we select the winning households, we update the household and building class variables, and store the changes *changelog*.

4.3.3 Building evaluation function

The choice behavior in the model is represented in the form of RRM$^{\sigma}$ allowing us to compare different housing alternatives. As discussed in the Section 3.5.1, we utilize an adapted version of RRM$^{\sigma}$ formula:

$$\boxed{R_i^{\sigma} = \sum_{j \neq i} \sum_{m} ln(1 + exp(1 \cdot [\frac{x_{jm} - x_{im}}{\sigma_{im}}]))} \tag{4.1}$$

The code form of this is captured in Algorithm 4.3. Important to mention, that prior to evaluation we first reduce the building alternatives (i.e. houses considered as candidates for a household). More specifically, we ensure that the people can afford the house and it has the minimal floor area requirements. This number is different for each municipality. Not being able to find one for the case-study area, we select The Hague as reference [64]: $12m^2$ for each person living in the house, with $24m^2$ for the minimal living unit floor area, defined in the building law of the Netherlands.

Algorithm 4.1: Run function

```
Data: Household table H, building table B, changelog table C, choice
      model specification M <string>, selection specification SM <string>,
      termination indicator T <list>
Result: Changelog table C
n ← 0 ; // step counter
// Repeat step for T[1] <int>
if T[0]=='steps' then
    for r from 0 to T[2] do
        H,B,C ← step(H, B, C, M, SM);
    return C;
// Termination change threshold:  moving average length T[1]
   <int> and threshold for that averageT[2] <int>
else if T[0]=='changes' then
    A ← 0 ; // Average indicator
    // Continue while the average below T[1] OR we have at least
       T[2] elements in changelog
    while A > T[1] or len(C) < T[2] do
        H,B,C ← step(H, B, C, M, SM);
        n ← n + 1;
        A ← mean(C[-T[2]:, 'changes']);
    return C;
// Termination by convergence:  defining moving average domain
   size T[1] <int>, offset for moving average comparision T[2]
   <int> and a safeguard of maximum number of steps T[3] <int>
else if T[0]=='convergence' then
    A ← 0 while True do
        H,B,C ← step(H, B, C, M, SM);
        n ← n + 1;
        A ← mean(C[-T[1]:, 'changes']) ;     // calculating the average
         number of changes from -T[1] to the last item of the
         changelog
        A_offset ← mean(C[(-T[1]-T[2]):-T[2], 'changes']) ; // calculating
         the offset average number of changes from -T[1]-T[2] to
         the -T[2] element of the changelog
        if len(C) > T[1]+1 then
            // if average the same, terminate (end) the simulation
            if A == A_offset then
                return C;
            // if number of steps exceeded, terminate (end) the
               simulation
            else if n>T[3] then
                return C;
else
    ModelError('Wrong limit inidication T');
```

Algorithm 4.2: Step function

Data: Households table H, buildings table B, changelog table C, choice model name M <string>, selection method SM <list>
Result: Household table H, building table B, changelog table C

```
for h from 0 to len(H) do
    if M == 'RRM_SD' then
        no_people ← H[h,'adults'] + H[h, 'children'];
        if no_people < 2 then
            area_threshold ← 24;
        else
            area_threshold ← 24 + (no_people − 2) × 12;
        B_filtered ← B, where B[ : , 'v_building'] < H[h,'capital'] AND B[ :
          ,'a_building'] > area_threshold ;
        R ← empty regret dictionary with keys as B_filtered indexes;
        R ← evaluate_buildings_RRM_SD(H[h], B);
    else
        ModelError('Wrong choice model specification M');
    if SM == 'ranking' then
        candidate ← key of min(R);
    else if SM == 'MNL' then
        probabilities ← exp(R[:]);
        total ← sum(R[:]);
        probabilities ← probabilities[:]/total;
        candidate ← numpy.choice(R.index, probabilities);
    else
        ModelError('Wrong candidate selection specification SM');
    bidding ← empty dictionary;
    if candidate != H[h, 'current_location'] then
        if candidate in bidding.keys then
            if bidding[candidate][0] < H[ h, 'capital'] then
                bidding[candidate] ← [H[ h, 'capital'], h];
        else
            bidding[candidate]=[H[ h, 'capital'], h];
foreach bidding.keys as k do
    household_id ← bidding[k][1];
    old_building_id ← H[household_id, 'current_location'];
    B[old_building_id, 'hh'] ← None;
    B[k, 'hh'] ← household_id;
    H[household_id, 'current_location'] ← k;
    get_household_properties(H[household_id]);
collect_to_changelog(C);
return H,B,C;
```

Algorithm 4.3: Building evaluation function, using RRM^{σ}

Data: Attribute value table $A(n_a \times n_j + 1)$, where n_a is number of criteria types, $n_j + 1$– number of alternatives, including reference; Criteria max/min objective list $A_O(1 \times n_x)$
Result: Regret value table $R(1 \times n_j + 1)$

```
for a from o to n_a do
    SD[o,a] ← σ(A[a]) ; // standard deviation of the attribute
     value list A[:,a]
    // If the values in the list are the same (no spread) or the
       values are NaN
    if SD[o,a] is NaN or SD[o,a] == o then
        SD[o,a] ← A[o, a];
        // If the values in the attribute list A[a] are all 0
        if SD[o,a] == o then
            SD[o,a] ← 1;
R ← table with n_j items as o;
for a from o to n_a do
    for i from o to n_j do
        R[i] ← o;
        for j from o to n_j do
            if j != i then
                regret ← log(1 + exp(A_O[a] × ((C[c][j] − C[c][i])/SD[a]));
                R[i] ← R[i] + regret;
return R;
```

4.3.4 Attribute value calculation

In our code, these attributes is captured in the *get_household_properties* function, which was also part of Algorithm 4.2. The variables and their calculation procedures are summarized in Table 4.1.

We classify the by types relating to properties of: (1) location; (2) location and household; (3) building. Additionally, we identify which of the criteria are static throughout the simulation procedure and which become dynamic. While talking about location related criteria, non-static implies, that the values are calculated for each alternative building a household evaluates. The only two non-static optimization criteria are the risk and real estate price. Risk is an ordinal scale attribute, which is either 1 (largest risk), 0.5 or 0 (no risk). The last, model run column shows within which model runs the variables have been incorporated, more on the subject in Section 4.4.

4.3.5 Output and communication

The output of the simulation process is captured in the *changelog* class. Throughout the simulation it is used to both capture the changes to the building stock, mobility of the households and overall system performance from the perspective of the evaluated criteria. Therefore, the definition of the code, as well as the class automatically adapts, based on which spatial, criteria and intervention extent is mapped. Given the criteria presented in the Table 4.1, the model outputs are:

Type	Criteria	Static?	Procedure	Model run
1	Distance to amenities	Yes	Range search and look-up from distance matrix (Algorithm A.3)	S, B
2	Distance to job	No	Look-up from distance matrix (Algorithm A.2)	S, B
2	Distance to school	No	Range search and look-up from FW matrix (Algorithm A.2)	B
3	Real estate price	Yes/No	Built in property	S, B
3	House area	Yes	Built in property	S, B
3	Parcel area	Yes	Built in property	B
3	Risk	Yes/No	Extrapolated from datasets & adjusted	B

Table 4.1: Criteria evaluated in the simulation for different spatial scales, corresponding to model development cycles (S: small, B: big)

System aggregate values of	**Notation**	**Units**
Distance to amenities	d_amenities	Meters
Distance to jobs	d_jobs	Meters
Distance to schools	d_schools	Meters
Real estate price	v_house	$10^3 \times$euro
House area	a_house	m^2
Parcel area	a_parcel	m^2
Risk	risk	unitless
Performance tracking		
Changes	c	steps
Changes per income bin	c_bin	bins: steps
Empty houses	vacant	units
Location history per change	loc_hist	household: start and end vertex

Table 4.2: Attributes given as the output for the model

Given this data, we create series of visualizations, which allow interactively explore the information. The maps and charts created provide statistical and aggregate summaries for all alternative runs, showing the average outcomes, relating to each of the interventions, described in Section 4.4.2). When we talk about model aggregate criteria, we refer to a summation of an attribute for all households in the simulation.

4.4 EXPERIMENTS AND INTERVENTIONS

4.4.1 Spatial extents

Within this research project we focus on two spatial extents (Figure 4.8): (1) a single statistical zone with a relatively small building and household counts; (2) A multiple statistical zone extents, which include large numbers of households and buildings.

Our initial runs are generated for the smallest administrative unit, called 'buurt', of the Dutch National Statistics Agency (CBS). The specific area we choose is Huizinge village. This area is one of the least populated 'buurten', but still has a wide selection of building functions (i.e. not only shops and residential buildings) and has a network with loops (i.e. streets that form rings) (Figure 4.9). The zone has 56 buildings, 48 of which have sufficient data to be used as model input. There are 45 households, leaving 3 buildings vacant. In this model setup we test the ba-

sic model functionality with 4 optimization criteria (see Table 4.1). Moreover, for this extent we generate 100 household sets, which are then run in parallel to create alternative area evaluations. The purpose of the runs is to verify model processes and outputs on a small, 'toy' scale. In essence, this scale is a scale up version of the hypothetical problem shown in Chapter 2. Just in this case we are dealing with 4 rather 2 optimization criteria and 20 times more households.

The second run encompasses 4 'buurten', which together form a 'wijk': level higher aggregation used in CBS. This subset has 1000 households and a similar amount of houses, which allows us to test the methods on a significantly larger scale. Additionally, in this run we start incorporating additional attributes to be evaluated (7 in total). For this extent we generate a 1000 household sets, which like in the toy problem are run in parallel for base and alternative runs.

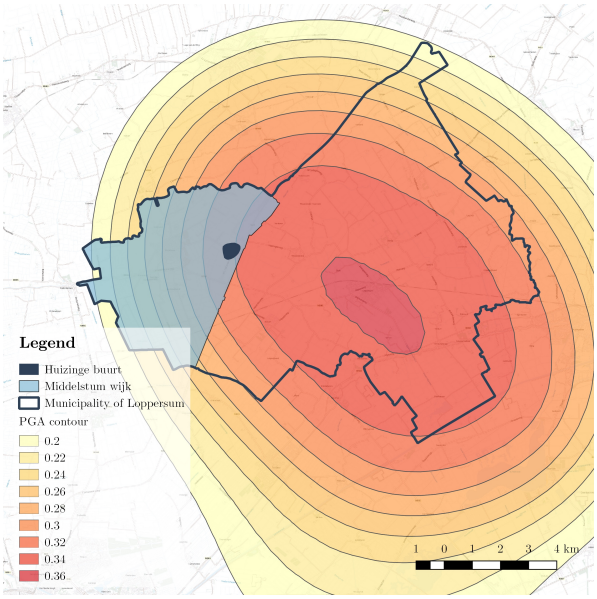


Figure 4.8: Subsets for experiments

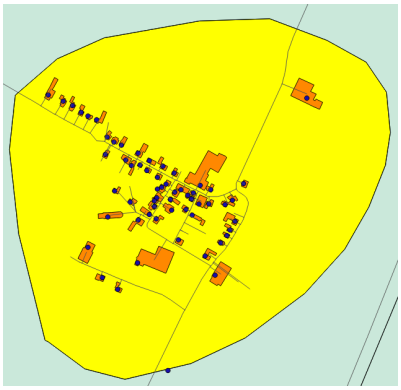


Figure 4.9: Preview of Huizinge 'buurt' with building, flat and network geometries

4.4.2 Interventions

The 'What-if' scenarios or the interventions created for this framework are required to showcase and test the usability of it. By selecting simplified, but common policy and spatial interventions, we seek two reach two goals. First, it allows us to create a platform of communication of model results. Secondly, it provides a medium to

find errors in the model definition and verify, that the optimization on agent scale is consistent also on aggregate scale. The cases that we choose relate to two ways of intervening in the simulation: (1) a change in input data attribute values and (2) the space representation.

Financial

For the first one we choose to inspect the impact on the simulated system by giving households financial subsidies. More specifically, we target the lowest income bins (based on histogram analysis with 'Sturges' and 'Freedman-Diaconis' estimators (see [153])) and provide them with subsidies for acquiring new buildings. For selecting the most effective amounts for subsidies, we generate set ranging from 5 to 70 thousand euro: [5, 10, 20, ..., 70]. In addition, we create a progressive subsidy experiment, where we attribute 3 lowest income groups a pre-defined subsidy. What is different from the static subsidy, is that we target groups with different subsidies. I.e. being in the lowest bin yields you a 3x of the subsidy, second lowest 2x and third 1x of the financial support. With both of these interventions we hope to identify whether the increased financial capability of the specific groups can lead to a better overall system performance (i.e. system-level aggregates for each of the criteria)

Structural upgrading

The structural upgrading scenario opens up the possibility for the model to become intrinsically dynamic. That means, that given specific conditions model entities (i.e. households) can change the properties of the space and themselves. More specifically, if a household identifies, that their current location is the best from what is available, they can choose to invest (i.e. reduce their capital) to remove the risk associated with the building and increase its' value. We assume, that the households have to cover 10% of the assigned upgrading cost, whilst the rest has to be covered by the government.

The primary purpose of this scenario is to showcase the dynamic capabilities of the ABM and inspect the effect of the preference weights (β) have on the upgrading counts. Therefore, we are not inspecting ranges of solutions like with the previous financial subsidies. However, we do change the preference weights for the risk attribute and observe the effects it has on the model outputs.

4.5 VERIFICATION AND VALIDATION

Firstly, to avoid common misconceptions verification and validation terms need to be explained. Validation is the "assurance that a product, service, or system meets the needs of the customer and other identified stakeholders" [52]. Verification- "the evaluation of whether a product, service, or system complies with a regulation, requirement, specification, or imposed condition. It is often an internal process" [52].

In a traditional sense, to verify a model means to ensure, that it represents a real-world phenomenon to a certain level. For this purpose we would first need to calibrate or estimate our model parameters, based on real behavioral data [74]. Nonetheless, by running the model on identical sets of households under different interventions, we can talk about model stability and sensitivity to specific inputs. This should be not mistaken with a sensitivity study, as this would require finding ranges of the model parameters (i.e. the inputs, β in the choice modeling) under which specific phenomena arise. This is not feasible within the current implementation due to long execution times of the simulation (see Section 5.3). This procedure is commonly followed model parameter estimation or calibration, not possible to

us due to lacking data on the observed choice behavior. Only then we can apply validation methods like back-casting [173] or Bayesian melding for uncertainty quantification [142]. Not having any behavioral data publicly available, the traditional validation falls outside of the scope of the project. Thus on the one hand, our primary concern during this research is to verify, that the simulation performs according to the expectation (i.e. it optimizes and converges). On the other– to validate, that the model provides sufficient information to be utilized in PS decision making process.

To verify the framework, we inspect the data processing and synthesizing outcomes (model input), as well as the modeling process and results. When possible, we inspect, the input data completeness and whether acquired spatial datasets still correspond to the non-spatial counterparts. Additionally, when the data is incomplete (see Section 5.1), we ensure, that the reduced modeling scope still has the same statistical properties as the aggregate input data. For example, even if a part of buildings need to be removed from the model input due to incomplete data, we ensure that the general vacancy rate remains the same. Similar tests and adjustments are built in throughout the code and when adapting data is impossible, it informs the modeler about the arising issues (for an overview see Section A.2). Additionally, we compare the model runs with sub-selections of evaluation criteria, which allows us to spot any further inconsistencies and exceptions in the code.

5 IMPLEMENTATION DETAILS

> We should forget about small efficiencies, say about 97% of the time: *premature optimization is the root of all evil*
>
> *Donald E. Knuth* [91]

In this chapter we will discuss the details and choices related to the model implementation. Firstly, we discuss the datasets, that play a role in the model definition. We go into detail how we extract the information and if multiple datasets are available for the same feature, explain the integration and/or selection criteria of them. Secondly, we provide backing for the software used and introduce the reasoning for not using ready-made modeling solutions. Lastly, we discuss the code performance and optimization endeavors.

5.1 DATA SOURCES AND PROCESSING

This section covers the details retaining to the acquisition and processing of the datasets for the case study. The structure of it follows the class definition presented in the Section 4.2.1.

5.1.1 Data overview, origin and retrieval

The data used in this model (Table 5.1) is predominantly open data (definition can be found in [149]). The datasets mentioned in the table have been integrated in the model, but not necessarily utilized– specifically, the public space feature was planned to be used as a optimization criteria, but was discarded to simplify the communication of the model. Among the sources of the datasets fall most Dutch key registers (Basisregistratie Adressen en Gebouwen (BAG), Basisregistratie Grootschalige Topografie (BGT), Basisregistratie Kadaster (BRK)), statistical data from CBS, network geometries (Nationaal Wegen Bestand (National Road Dataset) (NWB)) and spatial plans. These datasets are either available for download at the Dutch geo-portal [110] or as a Web Feature Service (WFS).

Additionally, we use 2 datasets, which are publicly available and free, but are more difficult to acquire. The first of the two is the real-estate values (Waarde Onroerende Zaken (WOZ)). The data is only publicly available, under the clause, that a single device can access 10 unique locations per day. Automatic harvesting of data through the service provided is not legal under article 40a of the "Wet WOZ" [132]. As the service provider is not willing to provide an extract of the dataset and we are prohibited to contact the municipalities directly at this point, we manually acquire the necessary extents of the data throughout the duration of the project. The second dataset of the two is Nederlands Regionaal Model (Dutch regional model) (NRM). This data can be acquired by directly contacting and following an extensive procedure of Dutch ministry of infrastructure and water management– Rijkswaterstaat. Nonetheless, the data is free of charge and available for research purposes.

Class	Feature	Source	Reference year
Buildings	Building footprints	BGT, BAG	2016
Buildings	Building functions	BAG, Host company	2016
Buildings	Parcels	BRK	2016
Buildings	Spatial plans	Ruimtelijkeplannen.nl	2016
Buildings	Building vacancy	CBS municipal level data	2015
Buildings/ households	Building population size	Host company	N/A
Buildings/ households	Building value (WOZ)	WOZ-register	2016
Buildings	Risk	Host company	2018
Buildings/ networks	Road network	NWB	2017
Buildings	Public space	BGT, features "begroeid" and "onbegroeid terrein-deel"	2017
Households	Aggregate household properties	CBS "Wijken" en "Buurten" dataset, "Buurten" level	2015
Buildings/ households	Zonal job origins and destinations, zonal workplaces and working populations	NRM, base year matrices	2014

Table 5.1: Datasets, their sources and classes they contribute to in the model

5.1.2 Building class

PARCELS The parcel dataset, BRK, is one of the key registers of the Netherlands just like BAG.BRK is a spatial dataset including the castrate boundaries [87]. The issue with the dataset is that it does not provide clear marking of different os of parcels. So our primary goal is to filter out anything, that would likely not be suitable for building a residential building. Additionally, we overlay this information with the Ruimtelijkeplannen (spatial plans)]) (RP) dataset, which allows us to select zones suited for residential development. This is all done in the step one of the pipeline discussed in Figure 4.2.2. The only detail missing in the description is the treshold for the circularity checks. After trial and error we choose 0.3. This value seems to return sufficient amount of parcels for simulation purposes, whilst removing any irrelevant road feature parcels. This procedure helps us identify 2 parcels suitable for residential building development in Huizinge 'buurt' , 51 in Middelstum 'wijk' and 239 in the municipality of Loppersum.

GEOMETRY The building geometries, that are used in this projects are taken from a Dutch key register BAG. However, for the model itself we only utilize the point geometries of the BAG verblijfsobjecten: smallest unit suitable for residential, business professional, or recreational purposes [133].

FUNCTIONS The building functions are used in the model to identify workplaces and destinations (e.g. schools, health-care, amenities) for the agents in the model. However, this information is covered only limitedly in the BAG [85, 131]. Therefore, we choose to utilize the dataset provided by the host company created using public datasets (see []).

Figure 5.1: The output of parcel processing

REAL-EXTATE VALUE As mentioned earlier, the real-estate value data is publicly available, but requires manual harvesting. Keeping in mind the time required to doing so, it is one of the reasons for limiting the project scope to a 1/3 of a municipality. Additionally, this data is not provided for each and every building and is rather consistent with similar types of buildings (i.e. farms, industrial buildings, monuments, to an extent flats). Given the fact, that there are no or limited data points to refer to per each type, we cannot extrapolate the possible value of this type and have to discard these buildings from the model scope. Noteworthy, is that the data is provided in relation to an address, which can be parsed and connected to a specific BAG identifier, allowing us to connect it to the building class definition.

WORKPLACES Jobs mobility and location information is one of the core pieces of this model. It is used to determine one of the optimization criteria for the agents– the job accessibility. Firstly, we find the job locations by comparing the day (P_{day}) and night(P_{night}) population information 5.1. If $P_{day} > P_{night}$, then a building can be tagged as a workplace. This is a crude approach, which could be refined by using the Landelijk Informatiesysteem van Arbeidsplaatsen (National Information System of Workplaces) (LISA) dataset [148]. This dataset is only available commercially and their free data only comes at municipal level.

For the smallest problem set (Huizinge neighborhood) we assume that all job openings in a zone are filled by people living in it. This is far from accurate (see e.g. [35])– the population does not necessarily work in the same zone as they live. However, most detailed freely available data on job mobility is of municipal scale [35]. This level is too coarse for the application at hand, as even the largest exercise does not go larger than a municipality. Therefore, to begin with, we implement the worker residence and job distributions based on arbitrary numbers. Having five zones ('buurten'), we create a 6x6 matrix (case + outside), which shows the proportions of workers coming from a zone (rows) and working in a specific zone (columns):

The total number of workplaces in the municipality in the reference year 2015 is 1600 and 1700 a year later [34]. The numbers whilst looking at the day/night population dataset show a unrealistically large 1840, covering only the case-study zones (1/3 of the municipality). This can be explained by: (1) we are not taking into account people working during the night; (2) the data is of a different year; (3) the counting methods differ.

Due to this we decide to look into the base dataset provided with the NRM. This data further underlies the fact, that the day/night population-based workplace

Zone	Day/night	Arup building tags	Workplaces
0	147	227	650
1	7	14	70
2	5	13	25
3	9	25	54

Table 5.2: Building tags as a workplace, based on two datasets in relation to workplace counts

counts are completely mismatched. The NRM dataset shows the data in a year earlier than we are looking into, yet the inhabitant number is roughly the same to the CBS dataset counts (3% difference). The working population is only 44% of the total population, 9,6% of which did not have a job. Additionally, NRM dataset identifies 799 jobs in the area, most of which are located in the Middelstum area (81%). When comparing the original and final tagging options (day/night versus Arup building tags, Table 5.2), we observe, that the original approach had a very small number of nodes in comparison to workplace number. Due to this, we discard the building level day/night data for this purpose and randomly distribute the workplaces, based on the building function tags (i.e. non-residential), through the nodes. This brings in another point of uncertainty in the model, but we accept it, as this data is likely to be accessible in real-world situation.

Later in the process, we attempt to integrate the data available from the NRM model input. The information that we are seeking are the base matrices, originating in the year 2014, which capture the numbers of trips, with their origins and destinations in zones [135]. The data also specifies the time of day for the trip (i.e. morning (7:00-9:00), evening peaks(16:00-18:00), rest of the day), mean of transport (e.g. passenger car, bike, public transport) and the motive for transportation. The latter aspect involves 3 motives:

1. Home–work
2. Business-related (i.e. home–business-related, other–business-related)
3. Other (i.e. Home–school, home–shopping, home–other)

For the specific application case, we extract the total trips of motive 1 in the morning peak hours. However, the data is only provided in 2 modes: truck and passenger car (driver) data. Keeping in mind the small spatial extents, the number of trips made by car to and from each of the interest zones was less than 5% per zone. This suggests, that majority of the trips are likely to be done by slow modes (e.g. bike, e-bike, foot). This data is estimated in the NRM itself, thus we discard the acquired trip counts for this project purposes, but suggest this line of inquiry for future work.

RISK As mentioned in the previous chapter, we integrate the risk values to Middelstum run of the model. The only available data on the risk is the selection of buildings, that require interventions. They are grouped in subsets or batches. However, within our subset there are only buildings of one of the upgrading batches. These batches are based on the prioritization principles presented in [2], which at its core focuses on as effectively as possible reducing the risk to human life. Specifically, the priority is given to the buildings with highest number of occupants and reinforcement of vulnerable building elements (falling objects and non-structural elements). From the 1175 buildings, only 62 (5.28%) are in the priority list. All of these buildings are within our inspection subset, as they have complete data.

Following the assumption, that the soil conditions and ground acceleration in the subset area are the same, we can extrapolate that buildings with similar inferred structural types are likely to be under the same risk. Within the subset we have 7 unique Global Earthquake Model (GEM) strings (structural type identifiers, for more see [28]) or 3 major structural types. Expanding our search to these strings allows us to identify another 607 buildings that could be potentially under risk (55.98% of the zone building stock) (see Figure 5.2)

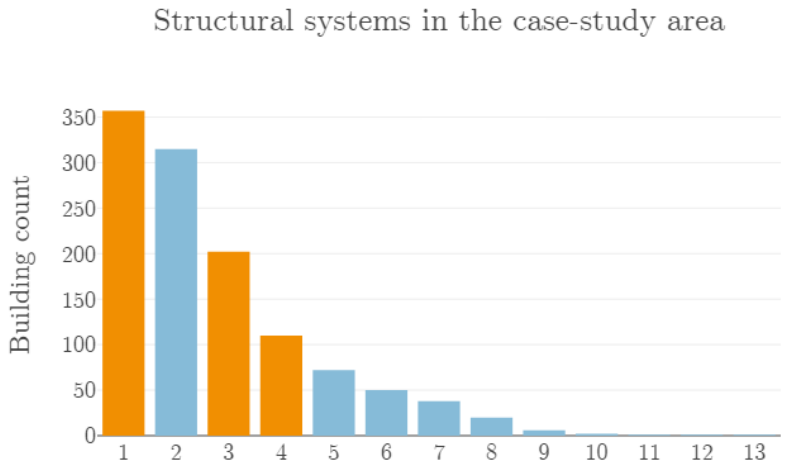


Figure 5.2: Building structural systems in the Middelstum case-study area; types in orange include buildings under risk

5.1.3 Network class

DATA SOURCE DILEMMA To be able to find the spatial relations between parcels and buildings, we need to constrain them to a network. The two main sources we are considering are two open datasets: NWB [136] and OpenStreetMap (OSM) [111]. When considering continuous space approach, another public dataset is added to this collection: BGT [86].

The NWB is owned by the Rijkswaterstaat (Dutch Ministry of Infrastructure and Water Management). The dataset is extensively used within the public sector. Among many examples of its applications are fast mode routing, traffic management, transport analyses for air pollution and accessibility, and infrastructure planning. The dataset includes all roads (thus also foot paths and bike lanes), as long as they have a street name [134]. Additionally, the NWB dataset consists only of line geometries, considered as valid for routing purposes. This implies, that the network does not have area features, the streets/roads are segmented only on the intersections, there are no duplicate geometries or orphaned (disconnected) segments (even though they can result from clipping).

The second dataset, OSM, primarily consists of volunteered data, but can incorporate open governmental datasets as well. This implies, that the quality of data is dependent on the area and can significantly differ from the governmental data (see **??**). The coverage of the dataset extends beyond the Dutch border, which would simplify the scaling of the method. This is showcased by a number of recent accessibility and environmental risk modeling studies (see [109]).

In contrary to the NWB, OSM includes roads like paths, bike lanes, steps etc. (for full overview see [112]). This is relevant while considering the modes we will be using in the model. The NWB dataset excludes considerable amount of connections,

which are meant for slow modes both on small (5.3) and large scale (5.4a). Nonetheless, on small scale most of them are insignificant, as they still fall within the same street representation as the motorized traffic. On the large scale, NWB dataset can be expanded using the Lange Afstands-Wandelpaden (Long Distance Walking Paths) (LAW) routes (walking path data) and the Fietsknooppuntennetwerk (bike path data).

However, the OSM data is not immediately usable for our purposes: the geometries are not of uniform type (i.e. only lines), segmentation is inconsistent, geometries overlap, have duplicates and can be orphaned (a French example is discussed in [67]). This would require significant effort to adapt the dataset to our purposes. Thus we select the NWB for the general network representation in the model, used to connect all other model features spatially.

All in all, we conclude that within the scope of the research area and for showcasing purposes, it is actually sufficient to use the NWB alone. Given the clear documentation and simple geometries of the dataset it gives no further issues and is processed as defined in Section 4.2.2.

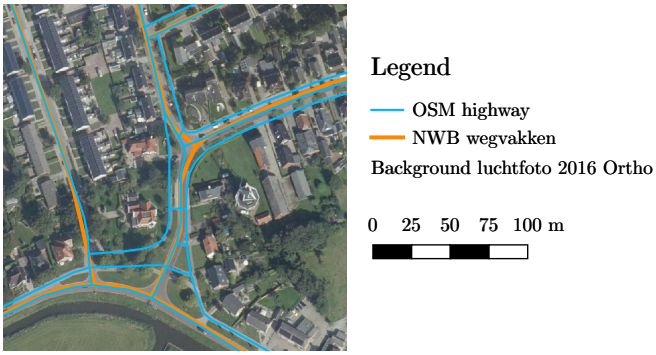
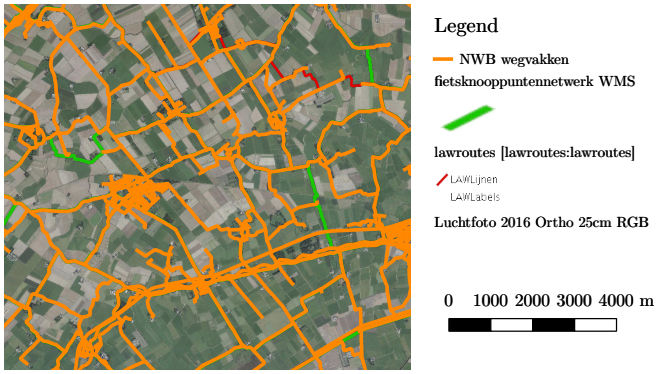


Figure 5.3: A comparison of NWB and OSM datasets, with an aerial picture in the background

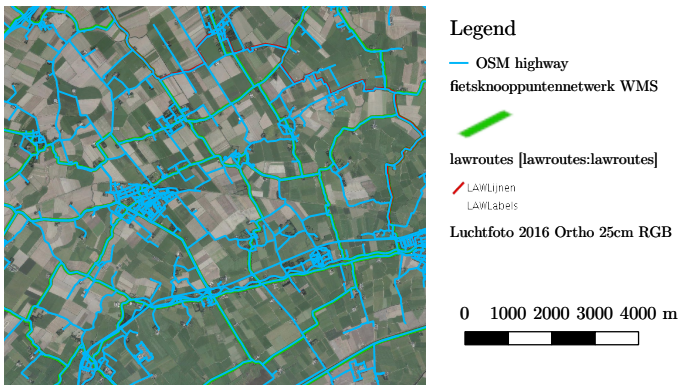
5.1.4 Households

The household data synthesizing procedure requires multiple datasets. The smallest scale demographic data available to us is "CBS vierkantstatistieken" (squares dataset)[33] and "CBS buurten" (neighborhood dataset) [32]. Both of the datasets are available through WFS, but requesting it through Feature Manipulation Engine (FME) trims the column names, making them unreadable (i.e. percentao5 or gemmidelo1). Therefore, the square dataset was re-acquired with the help of Atom feed. The resulting attribute names allowed to connect it to its documentation and constrain the population numbers to it. However, in the final application this data is not used. This is because of the limited coverage of the dataset (Figure 5.5), as well as the rounding of the numbers (i.e. rounded to multiples of 5 [33], p. 6). This is not the case for the higher level "CBS buurten". The harvested year is 2015. We choose this due to the attribute availability (see [33], pp. 40-43, [32], pp. 33).

WORKING POPULATION For number of workers, we take the total working population, taken from the NRM dataset in the case-study zones. This number is then reduced by the percentage of the unemployment rate in the region in 2015: 6.4% [34]. Additionally, based on the NRM zonal data, we define the proportions of population working outside of their zone of residence. For this we need to assume a number coming from external zones. Theoretically, the base matrices of NRM could be used to determine this, but, as mentioned earlier, they only have informa-



(a) NWB data



(b) OSM data

Figure 5.4: A comparison of network data overlays on bike- and footpaths datasets, with an aerial picture in the background

Origin/Destination	127	128	129	130	Total trips
127	6.8%	30.7%	6.4%	6.4%	2545
128	9.6%	25.2%	16.7%	8.8%	3453
129	7.0%	15.1%	5.6%	14.6%	2589
130	76.6%	29.0%	71.3%	70.2%	24721
Total trips	28468	1195	1232	2413	33308

Table 5.3: The result of the doubly-constrained gravity model for estimating the zonal dependencies between working populations and their jobs

tion on the personal car and lorry traffic counts. Knowing the totals for each zone, we utilize doubly constrained gravity model to find the numbers of people working in each of the zones. We assume that the travel costs are equal to one, as we are looking at a very small subset in space. However, it is potentially scalable by the travel distance between the centers of each zone. The results of the scaling is shown in Table 5.3

5.2 SOFTWARE

5.2.1 Acquiring and processing the data

We carry out data acquisition and pre-processing in FME Workbench [139]. It is an Extract, Transform, Load (ETL), flow programming software. The goal of utilizing this tool is to selectively harvest and merge the datasets, trim irrelevant attributes and reduce spatial extents. Additionally, FME provides built-in methods to produce

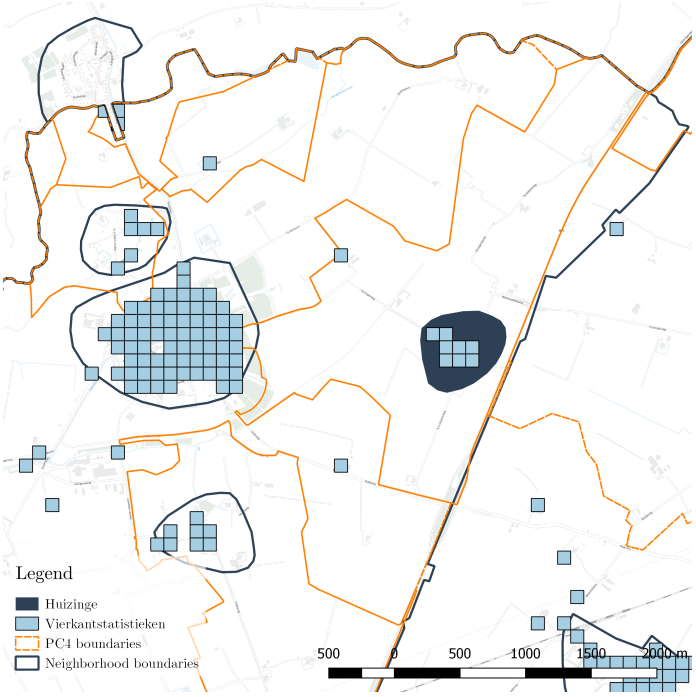


Figure 5.5: CBS dataset spatial extents and coverage

and test topologically valid networks. The latter is needed to calculate accessibility related criteria in the model. Although all of the datasets produced with this software can be spatial, this would create geometric redundancies (see Section 4.2.1). Therefore, the output of this tool also includes non-spatial datasets. However, everything produced is linked either by keys or spatial overlaps (see Section 4.2.1).

Noteworthy is that in this application we store the information in the local hard drive. And yet the sources of it are web-services and atom feeds. In other words, to bring the project data up to date one only needs to re-run the FME process: all its output is automatically stored in the project folders linked to the code for further steps. Nonetheless, the software also allows for creating automatic pipelines, seamlessly incorporating Python code with the native functions. However, this type of approach should only implemented in the end of the process. I.e. having independent feature modules allows for significantly faster design iterations. This signifies, that the pipeline feature is not essential for the current scope and thus is excluded from the project.

Python

After initial spatial and attribute reductions we choose to switch to pure Python coding. This allows a clearer control of the processes, reducing the module running time. The libraries used touch upon 3 subjects: non-spatial and spatial data analysis, and network processing.

The non-spatial data processing is carried out using Pandas. This library is selected due to its superior performance with large datasets, range of native methods to read and write the data. In addition, in the background it supports Numpy operations and allows almost seamless integration with it. This is particularly important when considering applying RRM, as the model requires multiple level for loops, some of which can be replaced with vectorized operations.

Within the model code itself, for spatial datasets we select the Geopandas framework, which in essence is a connection between Pandas framework and most common spatial libraries such as GDAL, Shapely and Fiona. During the later stages of

the model development process any spatial data preprocessing has been done in this framework rather than FME.

All graph related calculations have been carried out using an open-source Python library NetworkX. It forms an interface to code written in C, C++ and FORTRAN, allowing for relatively fast and flexible implementations. Moreover, NetworkX supports (un)weighted and (un)directed graphs, has built-in shortest path (Dijkstra, A* and Floyd-Warshall) and centrality related (e.g. betweenness, closeness) algorithms.

5.2.2 Modeling software

Just as extensive as the field of ABM is the selection of available modeling and simulation tools for it. Allan [7] gives a great overview of what is available in the field, with explanations for differing terminology. Yet in this project primarily 2 options were considered: NetLogo and Mesa.

The first of the two, NetLogo, is a "programmable modeling environment for simulating natural and social phenomena"[172]. Due to its active user community and the simplicity of use, some researchers see NetLogo as "the most promising language for becoming a standard in ABMs" [154]. Additionally, the software is open-source, has an extensive documentation and has seen many applications in geospatial domain (e.g. [114]). However, operating it requires knowledge of NetLogo programming language and, if extensions are needed, Java or Scala. However, linking with Python is also possible. For utilizing Python code directly in NetLogo, a Python extension was developed by Bryan Head [73]. In our case, this would be used to retrieve the attributes of the actors, while they change their location on the network. For controlling NetLogo from Python PyNetLogo library can be utilized [81]. This would allow us to start the model and retrieve its results. Nonetheless, this would require high proficiency in the modeling software.

The second option, Mesa, is an open-source (under Apache 2.0 license [8]) Python library. This toolkit allows the user explicitly handle the agent scheduling, integrates features observed in other ABM software such as batch-running (Repast, NetLogo), data collection (NetLogo). However, just like NetLogo, this modeling framework has its limitations. For instance, it only implements two representations of space: continuous and grid[100]. Additionally, the tools the library provides are rather self-explanatory, if ones would implement it themselves. For instance, move agents, get neighbors, run batch.

Given these considerations, we start implementing the code in pure Python code, but drawing inspiration from the elements of each of the libraries.

5.2.3 Visualization and interaction

Within Python there are 2 main libraries to be considered while making dashboards: Bokeh and Dash by Plotly. Other less used libraries include Bowtie, Pyxley and Pydashie. However, they either have very limited functionality/documentation or have been discontinued. In our research we choose to focus on Dash, as it not only is more effective with large datasets, but also more consistent and easy to use [72]. With this in mind, Dash sacrifices modifiability for user friendliness.

However, geo-spatial plots in Plotly still are rather limited. For instance, while utilizing Mapbox plots any custom polygon data requires a GeoJSON-based dictionary input. Each set of colors for polygons need to be loaded as a separate layer. This brings in a limitation, that only aggregate level areas can be displayed in such manner or large, discrete classes need to be used for visualization. Additionally, similar can be said about displaying the network: gradient visualization as if the network was overlayed on a continuous field is not feasible. Thus the coloring needs to be done in the same manner as for the polygons.

Regardless of the limitations, the Dash/Plotly framework allows for easy and fast prototyping, allowing us to show the basic functionality. Nonetheless, if such was to be brought on a commercial scale, whilst also allowing spatial user input, the logical step would be to continue developing using Geographic Information System (GIS) system as a framework (as proposed by [82]). This suggestion is based on reasoning, that these systems allow manual spatial input, more effective map styling options, as well as the possibility to integrate Python code in them.

5.3 PERFORMANCE

Performance plays a big part in the way the model can be operationalized. To be more specific, if we can perform analyses instantaneously, then a simulation can give real-time feedback to the user, who can define model inputs in a very responsive manner. However, in our case we are not dealing with linear, but quadratic computational complexity. This implies that the model times scales quadratically with the choice set size. In practice, we have observed similar tendencies.

The 'toy' scale model with 100 household sets had the runtime of 6 minutes, with small variations coming in due to different intervention inputs. Keeping in mind, that there were no dramatic differences between model runs, this is completely reasonable.

As expected, running the larger subset of the model with identical criteria, took significantly longer. 10 household sets took 3 hours to run and full 1000 household extents would have taken at least 12.5 days.

Thus in comparison, 45 households and 3 empty buildings versus 984 households and 56 empty buildings took 300 times longer to execute on average for a single run. Important to note, that the run times depend on the properties of both households and buildings, that become available. Due to the filtering done prior to the evaluation step, the larger subset runs have a lower amount of households, that evaluate and consequently relocate. However, given our current extents, the execution was so slow, that it required significant optimizations.

Keeping in mind, that fully optimized code is not part of our research scope, we remain utilizing Python. However, it becomes essential to make the model runs more manageable in the available time frame.

After code profiling (see [126]), we identify that the bottleneck of our code is indeed the evaluation function. It is the lowest level method in our code and is repeated in every step for each household and for each alternative house. Our optimization approach follows the guidelines in [117] and focuses on vectorization of evaluations. In other words, we reduce the layers of the core loop from 3 to 2, with the reduction coming in by performing the analysis on numpy vectors. Keeping in mind, that numpy is an interface to highly efficient legacy code, this allows us to reduce the execution duration to 1/10 of the original (Figure 5.6). Interestingly, even running the code on computers with significantly better specifications (see Section A.3), it had no influence of running times (see Figure 6.24). This suggest, that the further bottlenecks lie in data access, rather than processing. Nonetheless, this level of optimization is acceptable for the project purposes and spatial scope.

However, even with further improvements the code (e.g. compiling, interfacing to C or parallelizing) is very unlikely to perform at the required speeds for real-time feedback. This suggest, that the model application could potentially take the form similar to UrbanCanvas [150], where the computations are made in the cloud, many interconnecting models are pre-computed, but intervention feedback is not real-time.

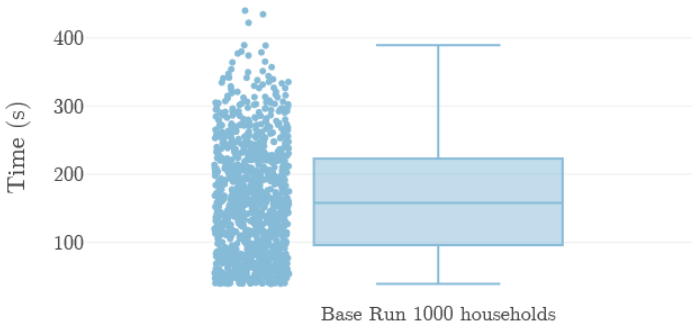


Figure 5.6: The run times box-plot for the base run with 1000 alternative sets

6 RESULTS

6.1 SYNTHESIZING

The synthesizing is the first step in the model and thus also a result of this project. Within this section we omit the 'toy' scale synthesizing results for brevity, as they are a simplified version of the large scale problem, but have similar properties.

To start of, the way we generate our data is not only based on the statistical counts on aggregate level, but is also adjusted due to the incomplete data on the building stock. Specifically, we randomly select to exclude specific households from the simulations to make the vacancy rates of the building stock according to the statistical data. Due to this the inputs to the model are not always identical. Firstly, the number of adults and children differs per household set (Figure 6.1). Secondly, the decomposition of the adult population also shows a distribution between different occupations. This underlies a property of the definition of our synthesizing procedure: the occupation is attributed post the reduction of the households, but the number of jobs remains the same. The reason why we see a large spread in the pensioner group is because this is the group that is identified the last and thus all the remaining adults, that do not have an occupation, are automatically marked as pensioner. The distributions of the rest of the occupations are also there. This is due to the fact, that the occupation is attributed randomly, but the adult population is smaller than the total of all the occupations combined. Additionally, this is not equal per zone we process, keeping in mind, that the working population can be also fully filled. To sum up, this aspect of synthesizing still complies to the definition of fitting to the statistical boundaries, keeping in mind the excluded households. Nonetheless, we do distinguish the possibility having it more controlled, if the occupations were attributed prior to the household exclusion.

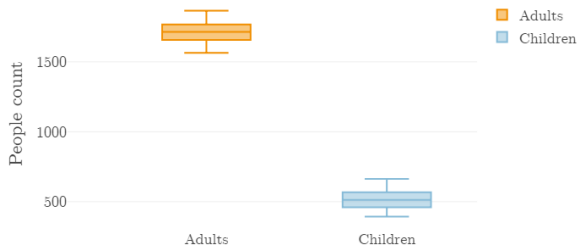


Figure 6.1: Population box-plot distributions adults versus children

The income representation within the household set can also vary. Namely, the households are randomly attributed to the building stock within the region (constrained to a region, such that the demographic statistics are still represented) and with it get the specification for their income (i.e. 110% real estate value). Keeping in mind the small number of vacant houses (i.e. 58), the differences between the income groups over multiple runs are minuscule, with the histogram boundaries being identical, according to the maximum of the 'Sturges' and 'Freedman-Diaconis' estimators (see [153]). The mean and the standard deviation boundaries of all sets can be seen in Figure 6.3.

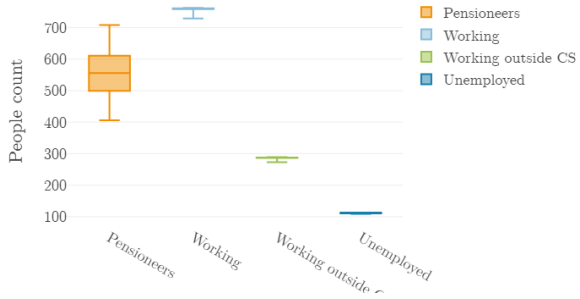


Figure 6.2: Adult population box-plot distributions by occupation, CS– case-study

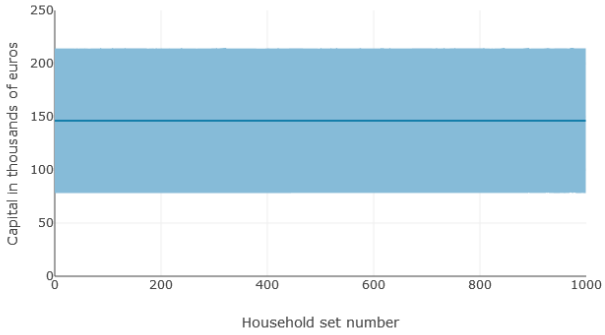


Figure 6.3: The mean and standard deviation of the capital of all households per synthesized set

6.2 'TOY' RUN: HUIZINGE

For the initial tests of the model we inspect a small case study area with 48 buildings and 45 households, whilst optimizing four criteria (see Table 4.1). The base run of the model on average would terminate, i.e. reach relative convergence around the 8th step, as seen in the Figure 6.4 in blue. The tail in the graph is due to the fact, that the model still did not account the oscillatory behavior arising due to several households exchanging houses between each other. This behavior is inevitable due to the small number of criteria available and the fact, that their properties were within the region, that the rejoice could outweigh the regret. This behavior was observed in a relatively large number of runs (i.e. 3 %), but is unlikely in larger systems, with more optimization criteria and wider range of houses.

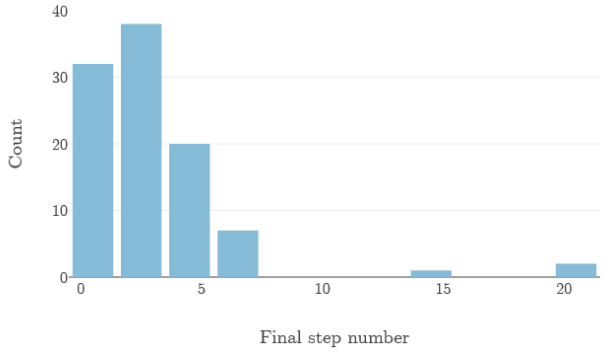
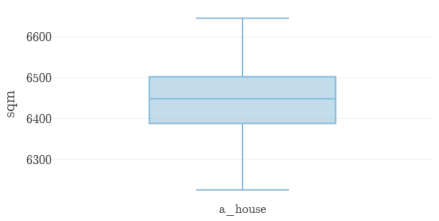
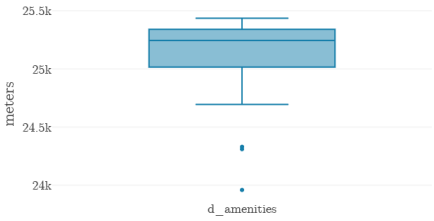


Figure 6.4: The number of steps before convergence histogram for the 100 synthesized household sets for the base run

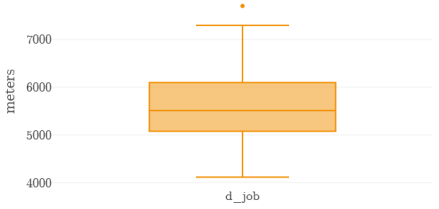
The base run, executed on 100 household sets had varying results. The boxplots in Figure 6.5 showcase the final step aggregate attribute values distributions. We observe, that 3 out of 4 plots do not show any clear outliers, but that is not the case for the distance to amenities criteria. This does not necessarily mean, that these specific runs are by definition outliers themselves. In other words, we need to see the attribute optimization as a product of all the dimensions (criteria), that we are looking at. As seen in Figure 6.6, there are no clear outlier runs, which could be discarded (as discussed in [56]). Moreover, it would be insufficient to to just look at them at the base run: same starting conditions should be inspected in the alternative, scenario runs and if the outlier behavior persists, identify the causes. In addition to that, the set of 100 alternative households is relatively small relating to the uncertainties of the household representation (i.e. no constraints imposed on household size and age composition during synthesizing). However, how big should a representative sample set be is a question for future study.



(a) House floor area



(b) Distance to 3 closest amenities



(c) Distance to jobs

(d) Property value

Figure 6.5: Distributions of final model step attribute aggregates for the Huizinge scale base run for 100 synthesized household sets

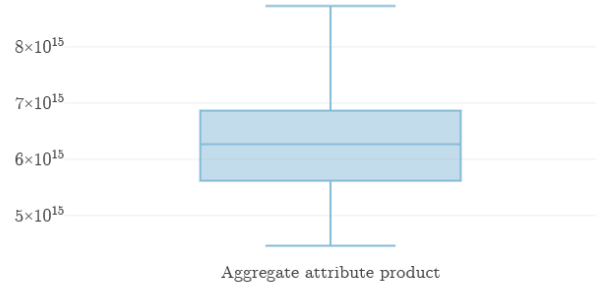


Figure 6.6: Product of all final model step attribute aggregates for the Huizinge scale for 100 synthesized household sets

6.2.1 Financial intervention

To further explore the model possibilities we propose a couple of hypothetical scenarios, which change the model inputs (explained in Section 4.4). In the simple problem we explore the possibility of giving the lowest income bin (13% (5-7) of people per synthesized set) a subsidy for relocation, as they are likely to be unable to improve their housing conditions due to financial constraints. This is even more accentuated due to our simplified bidding representation, where the household with the largest capital always wins. We provide series of interventions ranging from 5 to 80 thousand euro.

In our model specification households only move, if the alternative building causes them less regret. Therefore, we would expect, that the more movement we see, the better aggregate system performance becomes. Due to this, we first analyze the average changes per financial intervention (Figure 6.7). The results show, that the effect on the average number of changes (relocations) of the subsidy peaks at 40 thousand euros(+40k), which is in line with expectations. Namely, at this point, the two lowest income bins exchange places. The Figure 6.8 shows the +40k and base run aggregate changes per step, showing at which points the paths of the runs differ. Moreover, after decomposing the change behavior per income bin we see a clear increase in moves within the targeted lowest income group 1 (Figure 6.9).

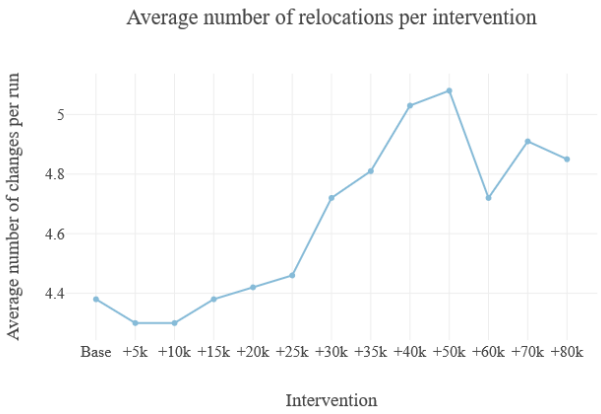


Figure 6.7: Average changes per step per subsidy (intervention) for the Huizinge scale

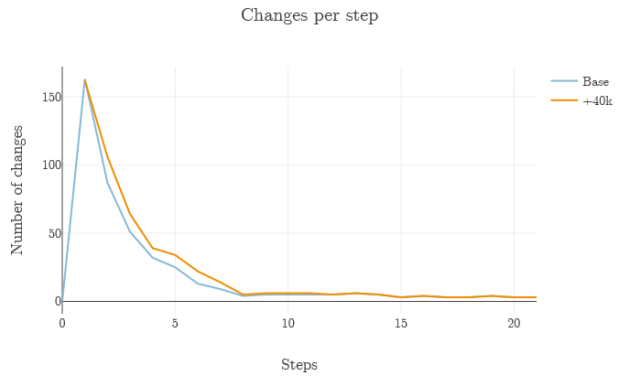


Figure 6.8: Total changes (relocations) per step for base run and +40k subsidy, given the 100 synthesized household sets for Huizinge scale

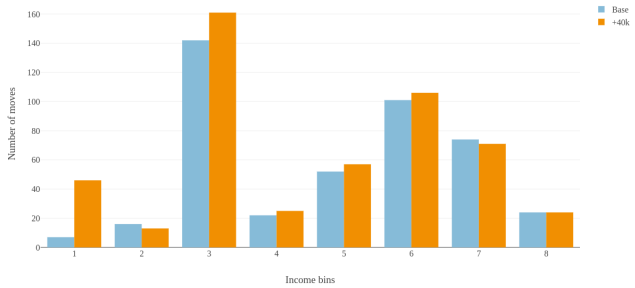


Figure 6.9: Number of changes (relocations) for income bin groups for the base and +40k financial run for 100 synthesized household sets

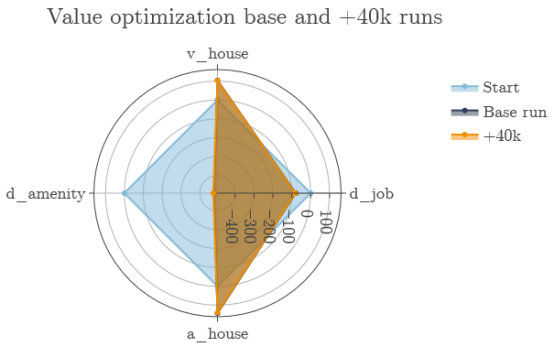


Figure 6.10: Criteria optimization for the base and +40k financial run

However, when we inspect the optimization per system level-aggregated attributes, we see completely identical criteria values (Figure 6.10). Even though, we are be possibly attaining better attribute levels for the lowest income group, it will not lead to better overall system performance. This indicates, that the primary issue with the observed urban system is not due to the fact, that agents cannot afford to move to specific houses, but rather the quality of the houses themselves. Namely, we are observing a system with mere 3 empty properties per each synthesized household set. This means, that in both runs the final result is that the same three properties remain vacant. Nonetheless, this sort of finding could potentially help identify buildings with unwanted attribute profiles. Of course, this is only valid, if the choice behavior is representative of the demographic in the area.

6.3 SCALING: MIDDELSTUM

For the scaled-up version of the simulation we have 984 households for each set with 1042 buildings as potential residences. We are evaluating 7 attributes (explained in Section 4.4). Based on the data collected for a 1000 synthesized household datasets (Figure 6.11, Figure 6.12) the model terminates after a median of 20 steps.

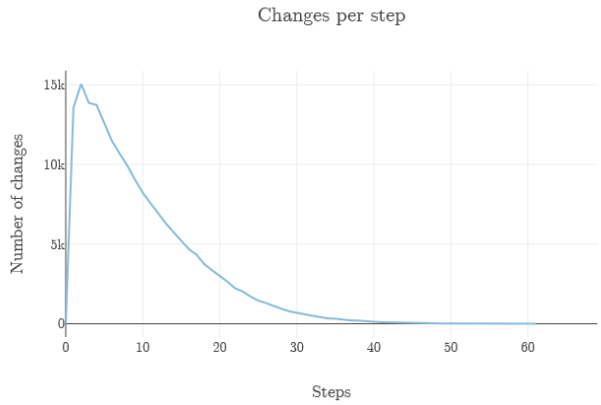


Figure 6.11: Aggregate number of changes (relocations) for each model step for all 1000 synthesized datasets for Middelstum scale

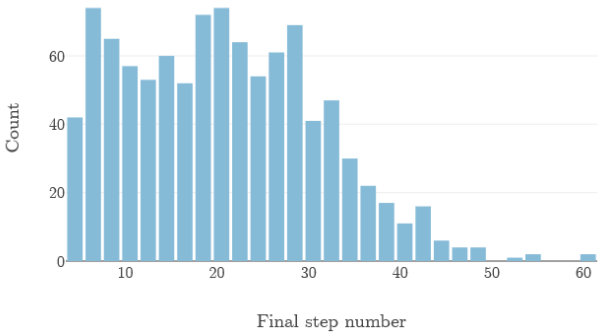
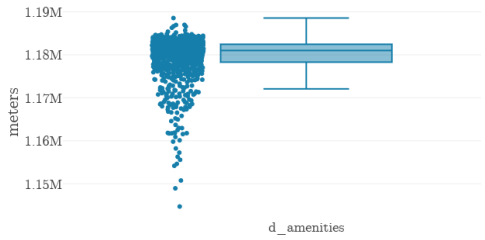
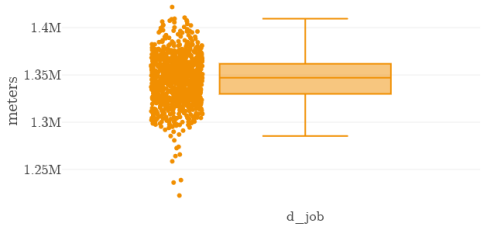


Figure 6.12: Histogram of the amount of steps it takes for the model to terminate. This data is taken over 1000 household set runs for Middelstum scale.

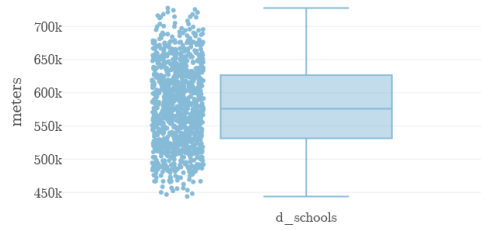
The aggregate attributes for the final step of the 1000 synthesized sets are not identical, with the distributions presented in Figure 6.13. Keeping in mind the larger sample set and the larger system, we notice, that outliers are more common among the different attributes. I.e. all of the optimized criteria, apart from distance to schools, had some outliers. However, when observing the product (subplot h), outliers are not evident.



(a) Distance to 3 closest amenities

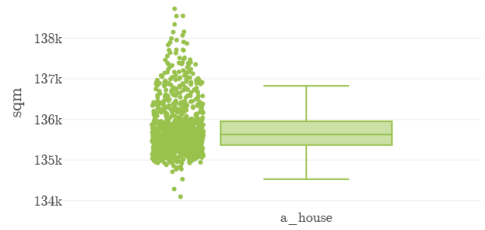


(b) Distance to jobs



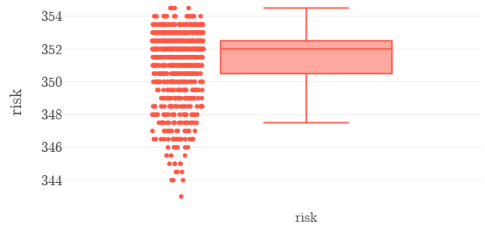
(c) Distance to schools

(d) Property value

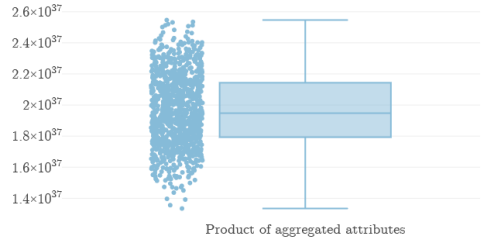


(e) House floor area

(f) Parcel size



(g) Risk



(h) Aggregate attribute product

Figure 6.13: Distributions of final model step attribute aggregates and their product (h) for 1000 synthesized household sets for Middelstum scale

6.3.1 Financial interventions

For the scaled application of the model, we are also enriching the showcase of the financial policies. In this subsection we will discuss two different approaches: static and progressive subsidy. Noteworthy is that these interventions are not run on the full extent of 1000 household sets, as that would have required around 11 days of runtime for simulating (see Section 5.3). Due to this, we run 100 household sets for each of the interventions.

Static subsidy

As explained in Section 4.4.2, static subsidy implies targeting one or more demographic groups and providing them with an identical subsidy. In the Middelstum case, we need to expand the the target groups from 1 to 1 and 2 lowest income bins, as the 1st bin has a median of 2 households for all 1000 household sets. With the 2nd bin added, this number rises to a median of 39.

After running the simulation, we observe very similar pattern to the small scale, Huizinge run Figure 6.14. Specifically, the average number of changes peaks at 40 thousand mark. After decomposing the changes per income bin, we get Figure 6.15. The image compares the impacts on the base, static and progressive subsidies (explained below). Looking at the first bin, we observe no relocations for the lowest income group, which is improved by the financial policy to 94 moves. The second bin is affected in a more extreme way: the number of relocations increases 5 times. However, most importantly, these changes do not come in significant detriment to other income bins. Namely, the first 18 bins increase their number of relocations, whilst the higher income groups are affected very minimally (+/- 5 steps).

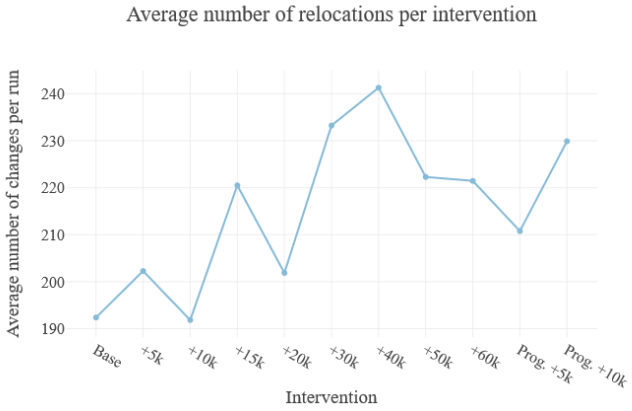


Figure 6.14: Average changes per step per subsidy (intervention) for Middelstum scale

However, we see little differences in the average aggregate attributes (Figure 6.16). In both cases we see the opposite of expected behavior in relation to minimizing for two variables: the risk and the real estate value. The first of the two is explained by the fact, that more than 50% of the building stock is under some risk. Therefore, if a household wants to increase any of the other criteria, they need to compromise (i.e. see the Figure 3.6). The increase in real estate value has the same nature. However, these compromises result in significantly increasing parcel and house floor areas, reducing the travel times to all the identified destinations. Between these two model runs, financial run exhibits better results (more minimization or maximization, depending on criteria) on all of the dimensions, apart from distance to amenities (slightly worse) and risk (identical).

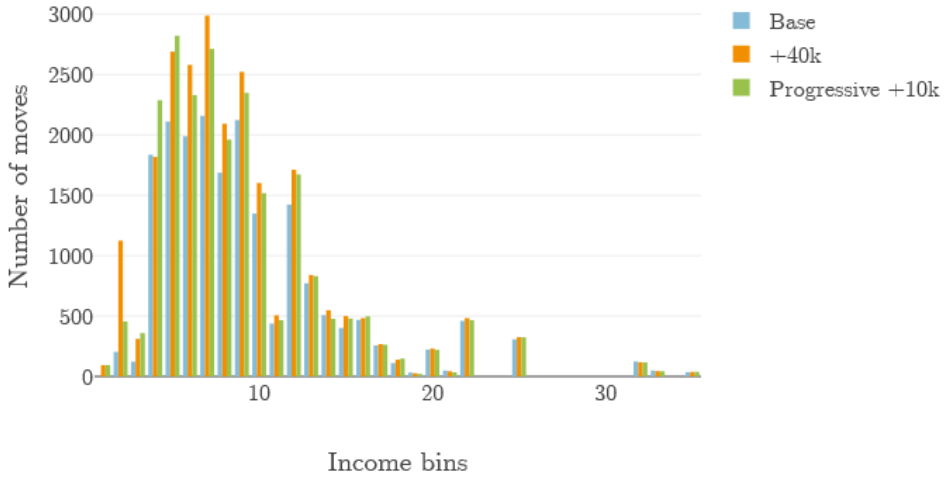


Figure 6.15: Total changes for all 100 household sets per income bin for base, static +40k and progressive +10k simulations

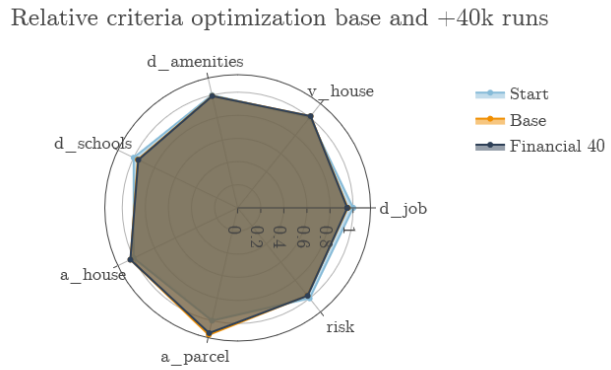
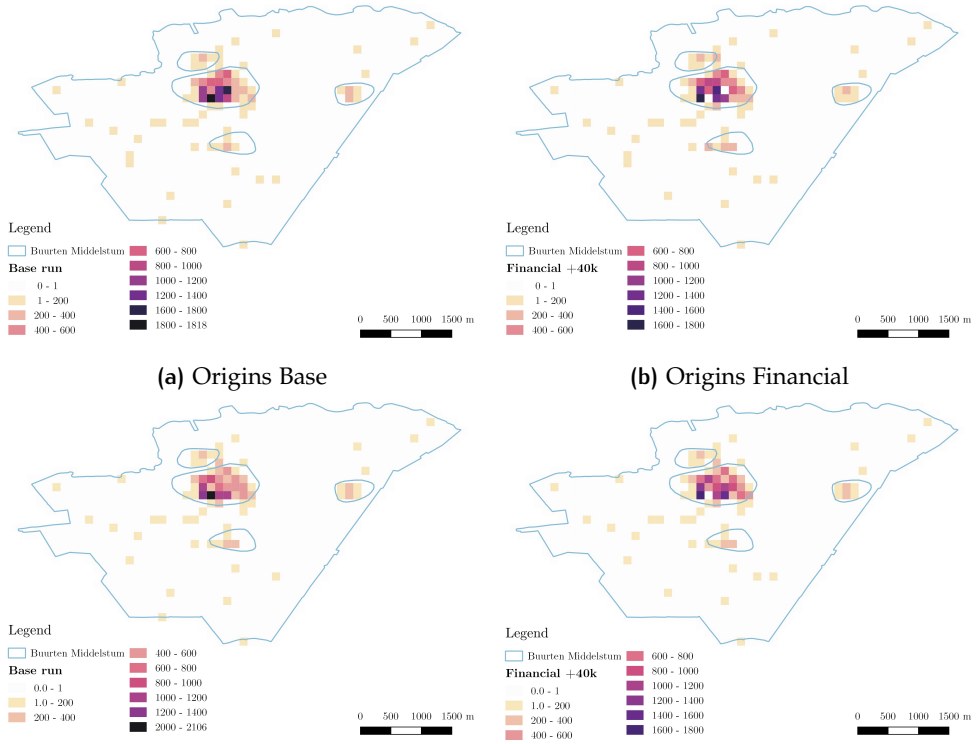


Figure 6.16: Relative aggregate criteria optimization in relation to the starting condition for base and +40k subsidy for 100 household sets, Middelstum scale

The policy also has impact on the spatial perspective. When looking at the full system relocations (Figure 6.17), we can see differences **from where** and **to where** all relocations happen. Notably, two of the most active origin (from where) tiles in the base run see no relocations in the financial run. This could be potentially indicate, that these tiles have lower quality (from our criteria perspective) housing, which is primarily used as temporary solution for a household until relocating to their final, most optimal house. Similar can be observed in the destination tiles. However, in this case only one of the active tiles in the base run sees no activity in the financial counterpart. This is also one of the earlier mentioned origin tiles. Seeing the other tile as an active destination tile suggests, that the earlier assumption of low quality housing stock is not valid for it. Overall, the patterns outside of the villages differ very little. However, we see at least one change the financial policy brings in. Specifically, one of the zones on the boundary of our case study area (southern border), is not a destination anymore. This suggests, that in the base run the households had to compromise the travel distances (majority of the destinations are in the villages) to improve other attributes. However due to new financial ca-

pabilities and freed-up building stock, these households were able to select a house closer to their optimum.

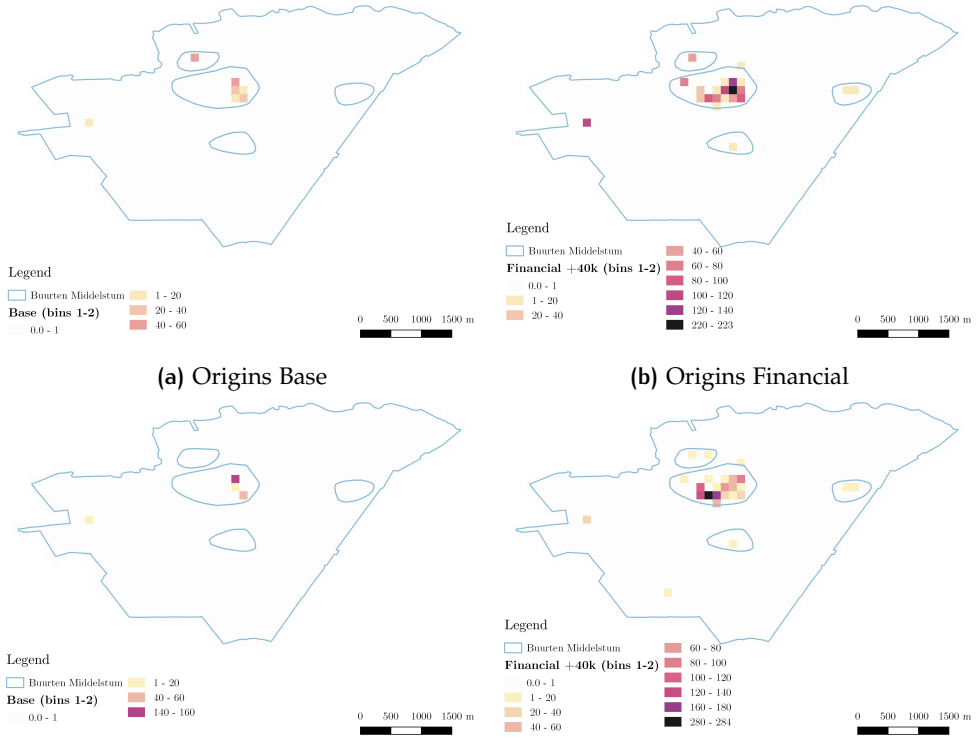
When decomposing the same runs in bins, we observe significantly different patterns for both origins and destinations of relocation Figure 6.18. We see more active tiles both in the periphery and the centers of the settlements. The spread to periphery could be caused by multitude of reasons, the simplest being, that there were simply more varied households moving (e.g. a family of pensioners without children would not optimize the 2 associated criteria).



(a) Origins Base **(b)** Origins Financial

(c) Destinations Base **(d)** Destinations Financial

Figure 6.17: The spatial patterns of the household relocations for a 100 synthesized sets in the base run and a scenario of a subsidy of 40 thousand euros for the 2 lowest income bins



(a) Origins Base **(b)** Origins Financial

(c) Destinations Base **(d)** Destinations Financial

Figure 6.18: The spatial patterns of the household (2 lowest bins) relocations for a 100 synthesized sets in the base run and a scenario of a subsidy of 40 thousand euros for the 2 lowest income bins

Progressive subsidy

The biggest difference between progressive and static subsidy, is than rather giving a constant sum to a specific income bin, we choose to make it dynamic. According to our specification, the lowest (and the smallest) income bin would get 3 times the investment, 2nd– 2 times, 3rd– the base-line subsidy. Thus with similar amounts of investment, you affect larger population. While comparing the changes of steps, progressive +10k (thus 1st group given 30k) intervention results (Figure 6.19) have similar average amount of steps as the static +30k subsidy (Figure 6.14). However, in relation with the exemplary +40k static intervention, it shows similar, but less extreme increase in number of steps against the base run (Figure 6.15). Keeping this in mind, in aggregate criteria we observe almost identical level of optimization in both financial policies, most of them being 1-2% better for the +40k intervention (Figure 6.20).

On spatial terms, progressive policy has very similar impacts as the static (Figure 6.21, Figure 6.17). However, as expected, the impacts to the two lowest income groups is less spread Figure 6.22 and having lower intensity values than the static. This does not exclude, that this policy still caused more movement in comparison with the base run.

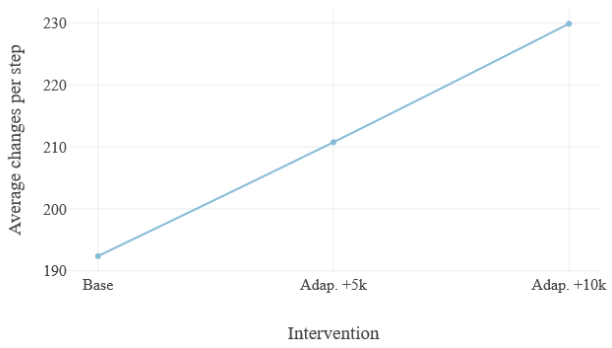


Figure 6.19: Average changes per step per progressive subsidy (intervention) for Middelstum scale

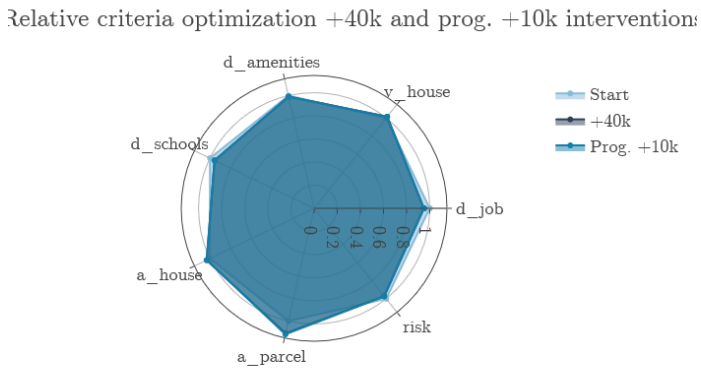
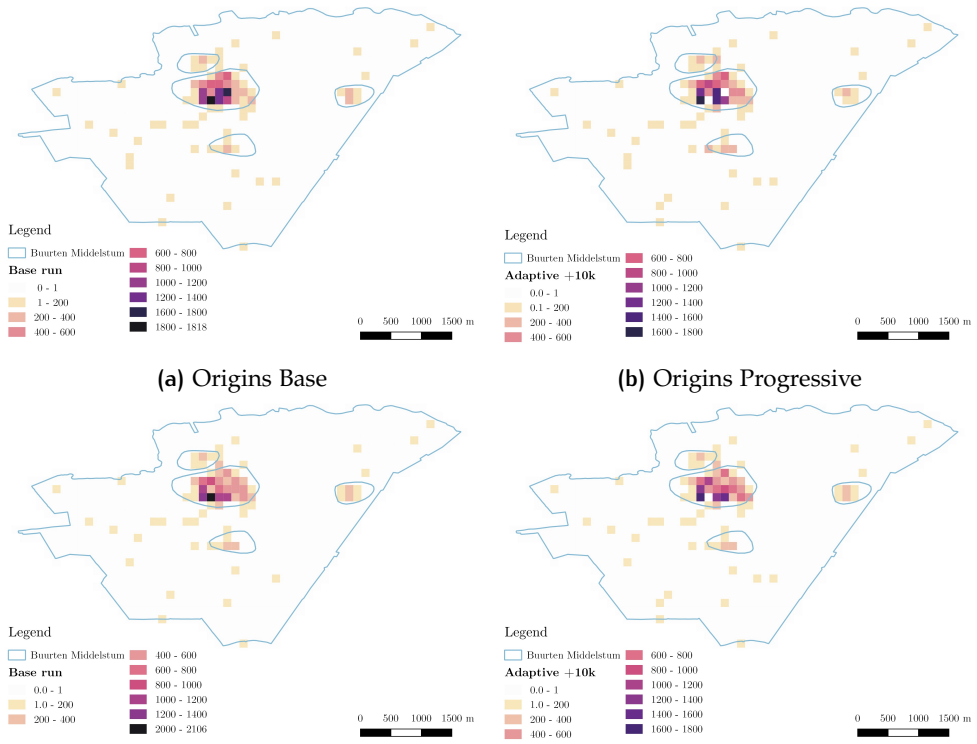


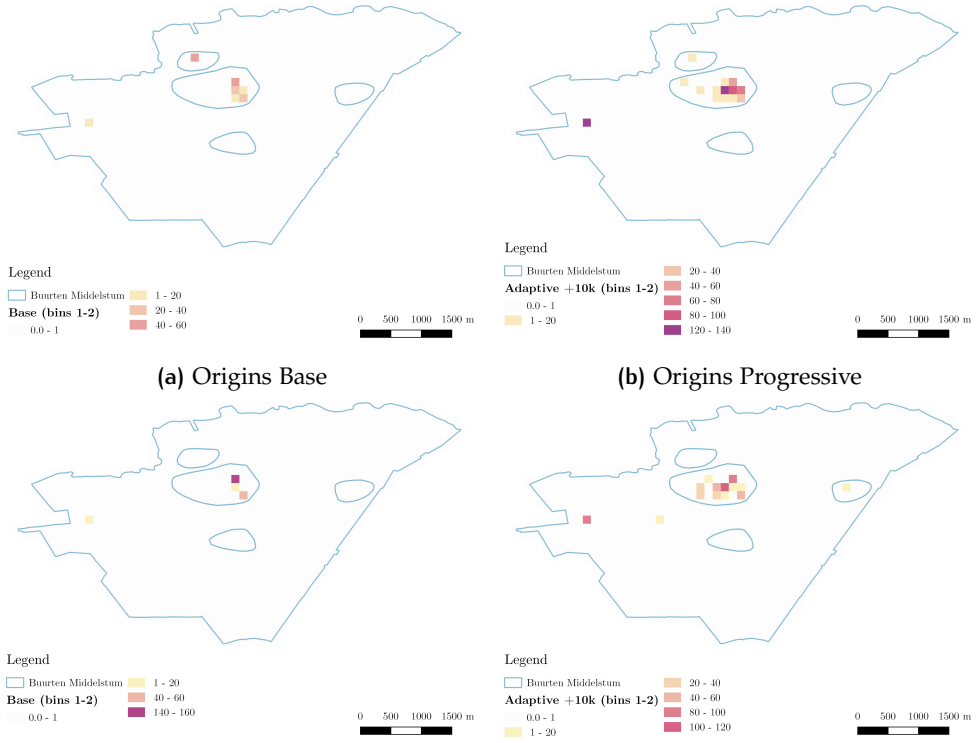
Figure 6.20: Aggregate criteria optimization in relation to the starting condition for progressive +10k (bins 1-3) and static +40k subsidy(bins 1-2) for 100 household sets, Middelstum scale



(a) Origins Base **(b)** Origins Progressive

(c) Destinations Base **(d)** Destinations Progressive

Figure 6.21: The spatial patterns of the household relocations for a 100 synthesized sets in the base run and a scenario of a progressive subsidy of 10 thousand euros for the 3 lowest income bins



(a) Origins Base **(b)** Origins Progressive

(c) Destinations Base **(d)** Destinations Progressive

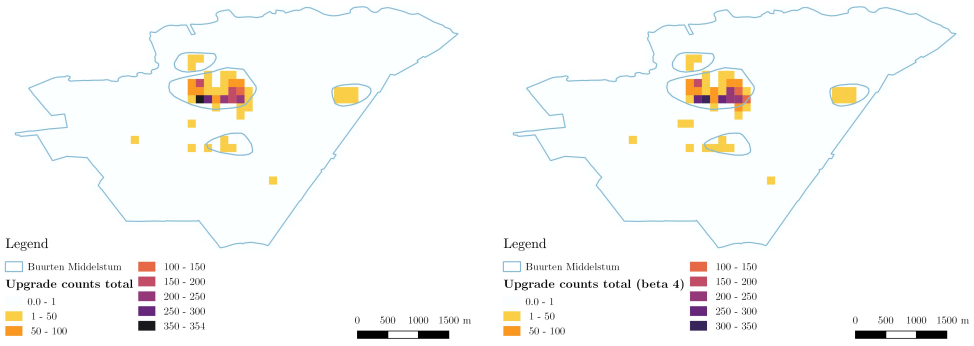
Figure 6.22: The spatial patterns of the household (2 lowest bins) relocations for a 100 synthesized sets in the base run and a scenario of a progressive subsidy of 10 thousand euros for the 3 lowest income bins

6.3.2 Subsidizing structural upgrades

The last type of intervention enables the agents to structurally upgrade their houses and so remove the risk, associated with it (details in Section 4.4.2). When comparing the final results of the runs with adaptive risk we observe very similar outcomes in relation to the aggregate criteria optimization (Figure B.4). In detail (Table 6.1), we can see some small scale effects on average aggregated attributes. If the risk preference weight β_{risk} is the same as for all the other attributes ($\beta_{risk} = 1$), we actually observe the opposite than expected effect on aggregate scale risk, i.e. it increases. This is because currently upgrades also reduce the capital of agents, which in turn effects their mobility. When the $\beta_{risk} = 4$, we see the average aggregate risk reducing, however the effects on all aggregate attributes are still very small (0-2%). From the perspective of upgrades, both preference weights return similar results (Figure 6.3.2). Namely, they are all clustered in the same regions and have almost identical upgrading counts (Table 6.1).

Property	$\beta = 1$	$\beta = 4$
v_house	0.0%	0.0%
d_amenities	-0.2%	-0.5%
d_schools	-0.6%	0.8%
a_house	0.0%	0.2%
a_parcel	-0.3%	-1.6%
risk	0.4%	-0.1%
average no. upgrades	22.81	21.1
$\sigma_{upgrades}$	9.31	10.13

Table 6.1: Relative difference between dynamic and base model runs average final step result



(a) Risk preference weight $\beta = 1$ **(b)** Risk preference weight $\beta = 4$

Figure 6.23: Number of upgrades per grid cell given different risk preference weights

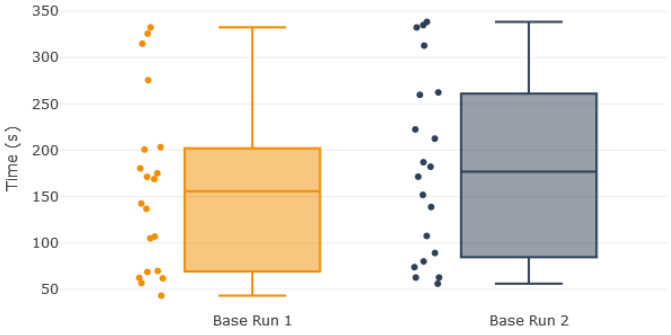
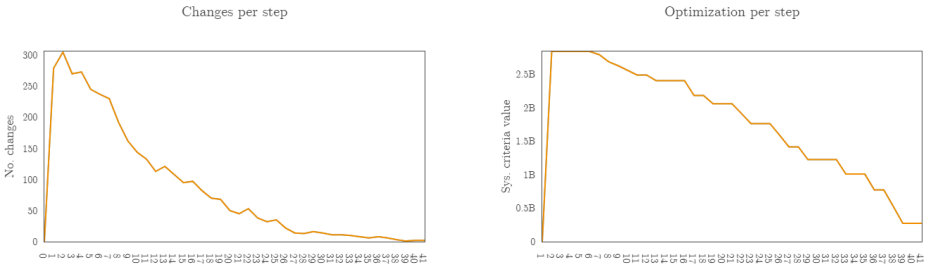


Figure 6.24: The run times boxplot for the base run with 30 alternative sets given two different machines



(a) Changes

(b) Optimization of system aggregate attribute product

Figure 6.25: Overlay of the comparative run results

6.4 STABILITY

One of the more important factors of our work is to prove that the model is stable and reproducible. To be more specific, that the behavior of the agents does not significantly change throughout the model. Namely, that given the same conditions and rules, model outcome remains the same. For this purpose we compare two model runs on the same subset of 30 households on two different machines.

The run 1 was executed on the secondary device, whilst run 2 was done on the primary computer (specifications in Section A.3). As expected, the run times between the devices was slightly different with the median for the run 2, being 25 sec. larger Figure 6.24. However, the distributions were similar, keeping in mind the small sample size.

Most importantly, the results and the result paths (i.e. performance per step) of both runs were the same. This was both on the primary indicator for differences, the number of steps, and the aggregate attribute product optimization (Figure 6.25).

7 CONCLUSION & DISCUSSION

> What matters is not "knowing everything" about the system in question but understanding the reasons and possible implications of our inevitable lack of comprehensive knowledge
>
> *Werner Ulrich* [155], p. 342

In this thesis we presented a computational framework, which allows simulating disaggregate residential location choice behavior. The chosen representation is built upon a notion, that people make decisions based on the choice set dependent regret, rather than context-independent utility. This is done by Random Regret Minimization (RRM) modeling approach. It allows capturing varying levels of regret (profundity) and enables incorporating multitudes of attributes of different dimensionality. However, the Random Regret Minimization (RRM) models have the estimated preference weight attribute unit scale dependent ($\beta = 1/unit$). Due to this, the preference weight would have to be re-estimated with each dynamic choice set, arising due to household heterogeneity and the dynamic vacancy rates in the housing market.

This is consequently applied on two different scale case-studies, situated in earthquake prone Loppersum municipality in The Netherlands. For the project purposes we are provided highly detailed building stock datasets, enriched with the information relating to the building risk. However, we lack micro-level population data, as it is commonly seen as private. As a solution, we showcase a method to synthesize the populations, based on several aggregate spatial and non-spatial statistical datasets. For each of the spatial scales, we create large numbers of synthesized household sets. They work as alternative model inputs and allows the modeler to identify the boundaries of the expected model outcomes.

Given these inputs, we construct a disaggregate, agent-based model framework, defined in a procedural, object oriented manner. This enables us to simulate micro level behavior of residential location choice. The agents in the model are the households making the decision to relocate and looking for a house. The heterogeneity of the agents is expressed in the household composition. Namely, it is conveyed in the dependencies to the workplace and school locations. The number of these relations depends on the household size, which is static, but conforms to the aggregate level statistics. The relations themselves are also based on zonal data, for which we utilize doubly-constrained gravity model to allow zonal worker inter-changes. The space representation is a mixture between discreet (i.e. there is a specific amount of houses to relocate to) and network (i.e. these houses are interconnected by a street network).

Furthermore, we showcase the simulation given 7 criteria for households to optimize for. These criteria relate both to the static properties of a building stock, as well as the household-related dependencies to the network. Given equal importance weights to each of the criteria, we observe stable and consistent model outcomes on each of the aggregated criteria, with convergence after a median of 20 abstract model time units (steps). The model capabilities are then further tested

using a series of financial interventions– subsidies targeting specific income groups. The model runs show outcomes in line with expectations and allow space related observations to be constructed.

Finally, this research allowed a definition of a working, but not yet predictive disaggregate model of residential mobility, based on regret. The aspect of disasters within this light has been integrated as a singular risk criteria, which has the mechanisms to be adaptive built in the model. However, given, that these interventions within the case study context are fully subsidized by the government and private agencies, we could not find a reliable, not 'ad-hoc' manner of quantifying the hindrance of the renovation. These aspects, would be essential to define if such a model would ever be operationalized in disaster research. Nonetheless, our hopes are, that this framework could enable integration of such findings in a transparent and systematic way, paving the road for a more constructive dialog between the parties involved.

7.1 RESEARCH QUESTIONS

The main research question on *"How to build computational framework that would allow to examine the residential choice behavior of households within a regional, disaster situation, given public sector agency-defined policy scenarios?"* can be answered in describing the process gone throughout the project. The formal name of the approach is the System, Theory, Modeling toolkit, proposed by Wilson [173].

In general, the procedure of making such a model starts with identifying the hypothetical or a 'toy' problem (Chapter 2). This allows one to draw the boundaries and define the system properties in a controlled manner. This consequently leads to defining the theoretical background and identifying the core methods and concepts. Given the fact that the subject of urban modeling is so vast, there is likely not only papers, but even open-source tools ready for your disposal. Even if one is to choose to write their simulation from scratch, as in this project, these tools are an invaluable guideline for development. However, the modeler should not underestimate the value of protocols, as shown in Section 3.4.1. These help a modeler define the framework and the core definitions of the model.

After drawing the outline, one needs to select a case study, and extract and process the data for it. Once this is done, an iterative process of creating more and more complex simulation begins. In our case we started by simply moving random abstract households into the houses that were empty. Once this worked, we added detailing on how these houses were chosen and what happens when two households intend to move to the same location. This is repeated until a sufficient model detail and scale is achieved. Of course, one also needs to define and communicate the outputs of the model, as well as be able to compare different model runs. This can be done by utilizing statistical methods or identifying performance criteria.

Verifying our model meant ensuring its stability and repeatability given the same starting conditions, as well as statistical checks during the data processing procedures. The calibration and validation are an essential step if one wants to describe the real world phenomena. However, due to not having access to data describing it, this step falls beyond the scope of a Geomatics master thesis. The details relating to the specific choices taken during this process can be approached by answering the sub-questions.

How to abstract housing choice behavior of households in a disaster situation on a regional scale?

For the first one, we need to differentiate between individual and group choice behavior. The individual is represented by a RRM$^{\sigma}$ model, which assumes, that

households base their decisions on context, i.e. choice set, dependent regret. The collection of such individuals making decisions are then integrated in an agent-based model. Here the entities, or agents, making the decisions, are the households choosing a residential location. The disaster risk aspect is represented as a property of the building stock, which can improved, if a household invests a part of their capital.

What modeling approach would be suitable for such abstraction, given a PS planning or policy making process?

We model the system as a collection of agents, which allow for bottom-up optimization processes to happen. The choices of the agents is represented by utilizing regret-based discrete choice model. In the context of Public Sector (PS) planning or policy making process, this simulation needs to allow testing 'What-if' scenarios. In our case, these interventions are represented as rules, allowing to change the input data or enabling the simulated entities to alter this data during the simulation (i.e. upgrade the building stock, given some investment).

What (type of) data could be used to generate model criteria values and what is its relation to the data available for the case-study?

The third question relates to the RRM^{σ} specifications, which requires to have quantified relations between different options. Apart from that, it can deal with both ordinal, interval and ration scales. Nonetheless, whilst talking about residential location choice, the inputs in our framework should be contained within the class definitions. Namely, it should relate to the building, parcel, household properties or intervene in the space (i.e. change the geometry of the network by adding new nodes and edges). However, in relation to the data availability of case-study, we are constrained to aggregate level population statistics, public spatial datasets and risk definitions provided by the host company.

What output should the model have and how to communicate it to the PS organization given a disaster mitigation or preparation situation?

Fourth, the model outputs a variety of properties. They include aggregate of each optimization criteria values for all households per model discrete time unit (step). Moreover: the relocations happening per step, both in relation to the household that relocates, as well as the income bin it belongs to. And finally the building stock, that is occupied and made vacant. Our primary communication methods are plots, representing these non-spatial units. However, each of the model entities are connected to the space, allowing us to create both maps (like the heat-maps presented in the results section) and animations, showing model changes per time unit.

What are the uncertainties and limitations of the model how to circumvent them?

Fifth, the uncertainties in the model are twofold - epistemic and aleatory (see Table 3.2). The first is associated with the fact that we have limited knowledge of the system attributes. Such is the results of the household synthesizing, risk definition, job location attribution. These uncertainties are possible to circumvent by collecting additional data and refining model specification. The aleatory uncertainties on the other hand are significantly harder to avoid, as they are caused by the intrinsic variability in nature. Such are the uncertainties in the actual choice behavior of households in the region. This can be to some extent accounted by estimating the model and refining the demographic subsets that exhibit specific choice behavior. Even with this in place, there is a chance that on individual level the choice behavior would not be rational. Due to that, it is essential, that models such as showcased

in this research incorporates error terms, allowing for such behavior. This stochasticity is inevitable and that is why many agent-based applications have batch runs implemented, allowing for identifying outliers.

What type of data would be necessary in order to calibrate such a model?

Finally, to continue with the calibration of this model, one would first have to perform analysis on the choice behavior in the region. Namely, it is essential to identify different demographic groups exhibiting similar choice patterns and ensure their representation as heterogeneous agents with individual taste parameters in the model application. However, the actual decision procedure is extremely complex (see Figure 3.3). Additionally, it is hard to estimate the amount of different relocation options a household considers. This would indicate, that the model would have to rely on stated preferences for the initial calibration. Next, one could consider using back-casting (i.e. applying the model in past situation and then validating it's predictive abilities for a point later in time) and sensitivity analysis for further refining the definitions of the taste parameters.

7.2 DISCUSSION

7.2.1 On simulation and results

The simulation we performed showed relatively stable results and expected outcomes in relation to optimization objectives. However, we concur with Lee et al. [94]: even with all of the opportunities ABMs open up, their output analysis still remains a large challenge. More specifically, we observe the need to perform more rigorous statistical analyses on the outputs, but also sensitivity analysis.

These challenges of testing and verifying the simulation results stem from the code performance. Specifically, a single set requires a mean of 4 minutes run time. Due to this, we are restricted to testing very limited model inputs, such as only 100 household sets per each intervention. Moreover, the way we inspect the model output variance is very rudimentary. We exhibited the error distributions of each of the attributes and all attribute products for the final step of each alternative household set. However, to fully understand the (dis)similarities between each of them, we would have to quantify, how many attributes exhibit outlier behavior for each of the model alternative model runs. This is done by quantifying the standard error for each of the synthesized sets in every scenario. Nonetheless, given, that the model is not calibrated, more extensive studies should be carried out to identify how the model parameters influence the parameter space (see Figure B.3).

Additionally, we have many simplifications in our models. For instance, within the project we ignore the unobserved part of regret (the error term). This is done explicitly to verify the model stability, but should be integrated in the future iterations of the work, once the estimation procedures are done. On a similar note, we assume, that people are completely rational and always choose the option causing the least regret. For a more accurate representation, one would have to implement a Multinomial Logit (MNL) model, explained in Section 3.5.

7.2.2 On risk and upgrading

In this case study, we strove to showcase the application of risk as an adaptive element in the model. This enables the end user pose questions relating to the cost of an intervention not only from a subsidy, but also upgrading perspective. Yet, the representation and impact of disaster risk in this model is still very limited. This because the available data is incomplete and contains only the address tags

for buildings identified as being under 'high' risk. The rest of the information is assumed based on the buildings' likely structural type [41]. This could be improved by incorporating the risk model into the framework.

Another aspect to consider, is that the upgrading costs are also randomly generated, based on expert opinions. Moreover, we generate this data only once, rather than testing the effects of it on the model outputs. This could be done better by integrating more complete information, datasets on upgrading costs in the region. This would allow us to identify specific upgrading cost associated with structural types in the case-study area. This would consequently allow to more accurately simulate the likely effects of the 'What-if' scenarios.

7.2.3 Usability aspect

The usability of this framework was one of the core objectives during the development. This is the reason for primarily relying on public datasets, since many of them (e.g. zonal statistics) are available outside of the Netherlands. Their processing pipeline is likely be similar and thus re-usable from this project. However, the aspect of data-hungriness remains a big challenge: with more detail needed in the model, the more data is required. Even though we managed to create a working model with only 10 datasets, this number is likely to rise if the project is developed further.

Additionally, we also paid attention on the code extensibility and the possibility to integrate with other models. Namely, the scripts were written in an object-oriented manner, clearly identifying the entities in the model. Moreover, all of the scripts were written in Python, which is known for the ease of learning and development [165], and is at the core of one of the most advanced general urban models UrbanSim. Since the code of the latter is open access, this opens a door for future integration.

However, as the main research question suggests, in this project we also strove to showcase the applicability of such models in public planning and policy making context. Given the hypothetical scenarios, we approached this by providing the results in a series of interactive visualizations, part of which were provided in Chapter 6. The main strength of this model lies in its flexibility to integrate variety of different types of data, as well as present it on aggregate and dis-aggregate scales. Given that the model is calibrated and represents urban phenomena to a degree of certainty, it would allow to inspect the effects of policies in a controlled laboratory. To go even further, if the issues of performance (Section 5.3) are overcome, such a model could potentially form a basis for serious game in the spirit of SimCity [107] or more recent Cities:Skylines [120]. With this vision we would see such a tool educating, allowing to interactively determine the scenarios and explore the complexity of the data.

7.3 FUTURE WORK

Integration

There are several directions, that this work could go further. Firstly, whilst developing further, one needs to consider integration within other modeling frameworks (e.g. UrbanSim [166]) or extended by other urban sub-models. The first attempt to make it integrable was to script in an object oriented manner, following the principles laid out in protocols (as Section 3.4.1) and existing tools (such as MESA [100] or PyNetLogo [81]). We would like to provide 3 initial directions for identifying the integration possibilities.

To begin with, excluding land-market representation in similar models as presented in this work could potentially lead to biased results, state Huang et al. [77]. In their extensive study, they provide many model examples and their best features. Moreover, the presented model only inspects the home-owner market. For a full picture one should strongly consider inspecting the aspects private rent and housing corporation markets.

Another type to consider is job relocation model: residential and job re-location timings are often correlated [129]. Such integrations have been shown in the works of Rashidi et al. [128], as well as Acheampong [4].

Thirdly, we would like to suggest explicitly incorporating a risk model and not only it's results to the residential mobility model. This last aspect comes in line with the initial motivation for this work and recent efforts in housing market research (such as [46, 47]) show the growing interest for it in the scientific community.

Dynamics: disasters and life-cycle

The more realistic disaster representation in this sort of model also touches upon the subject of dynamics. Currently, the model does not represent real-world time. However, a step towards that would be identifying realistic transaction costs for the processes the households are performing (e.g. relocation, renovation). Another thing to consider in relation to long-term processes, is that life-cycle events are one of the major contributors for the decision to move [4, 5, 77]. These events include changes to socio-demographic attributes such as age, household composition, job location, shopping patterns and capital. Their representation would allow population to change dynamically. However, the necessity of such sub-model should be evaluated in the light of a specific application case.

Heterogeneity and calibration

The Agent Heterogeneity (AH) should be refined based on the observed choice behavior. Namely, the first step towards this would be to identify whether there are different choice patterns between demographic groups. One way to approach it is Latent Class models. They enable identifying "the probability that decision maker n chooses alternative i, which equals the sum of the probability that he belongs to class s multiplied by the probability that i is chosen given the class s "[157]. Given this information, agent heterogeneity should be represented in the model as specific household properties, allowing one to identify the correct decision rules. These rules are the specifications of the RRM model, that are calibrated, based on the choice data (for estimation code see advancedrrmmodels.com). More suggestions on AH can be found in [77].

While gathering data for AH definition, one should consider retrieving information on the housing search too. In other words, the number of considered alternatives should be one of the properties differentiating households. This is important, as not all demographic groups adopt the same residential location search strategies.

After calibration it is important to identify the uncertainties of the model definition. One of the approaches is the Bayesian melding method [142], applied on UrbanSim model. Other approaches include Generalized likelihood uncertainty estimation [21], which is similar to Bayesian melding, but the likelihood function is derived from 'goodness of fit' to observed data. Given the quantities of interest distributions attained by these approaches, the model runs falling under the threshold can then be removed with more observations becoming available [56].

Usability

The usability of the model could be validated by incorporating the visualizations in a dashboard, which would allow public sector agencies to interact with the data.

By performing series of semi-structured interviews, future researchers could identify missing features in the model and properties to collect during the simulation. Making the model easy to understand and use would make the model more likely to be operationalized [66, 167].

BIBLIOGRAPHY

[1] Hazard and Risk Assessment for Induced Seismicity Groningen. Technical Report November, Nederlandse Aardolie Maatschappij B.V., 2015.

[2] Meerjarenprogramma Aardbevingsbestendig en Kansrijk Groningen 2017 - 2021 | Beleidsnota. Technical report, Nationaal Coördinator Groningen, 2016.

[3] UrbanSim — urbansim 3.1.1 documentation, 2017.

[4] R. Acheampong. Understanding the Co-emergence of Urban Location Choice and Mobility Patterns: Empirical Studies and an Integrated Geospatial and Agent-based Model. Technical report, 2017.

[5] R. A. Acheampong and E. Silva. Land use–transport interaction modeling: A review of the literature and future research directions. *Journal of Transport and Land Use*, 8(3):11–38, jul 2015.

[6] D. E. Alexander. *Principles of Emergency Planning and Management*. Terra Kunuted, 2002.

[7] R. Allan. Survey of Agent Based Modelling and Simulation Tools. Technical report, STFC Daresbury Laboratory, Daresburry, 2011.

[8] Apache. Apache license 2.0, 2004.

[9] S. M. N. Arifin and G. R. Madey. Verification, Validation, and Replication Methods for Agent-Based Modeling and Simulation: Lessons Learned the Hard Way! pages 217–242. Springer, Cham, 2015.

[10] Arup. SPeAR (Sustainable Project Appraisal Routine).

[11] J. Barthelemy and E. Cornelis. Synthetic populations: review of the different approaches. Technical report, 2012.

[12] M. Batty. *Urban modelling : algorithms calibrations, predictions*. Cambridge University Press, Cambridge :, 1976.

[13] M. Batty. Fifty Years of Urban Modeling: Macro-Statics to Micro-Dynamics. In *The Dynamics of Complex Urban Systems*, pages 1–20. 2008.

[14] M. Batty. Urban Modeling. *International Encyclopedia of Human Geography*, pages 51–58, 2009.

[15] M. Batty. Cities, Complexity and Emergent Order, 2011.

[16] M. Batty. Agents, Models, and Geodesign. *ArcNews*, 35(1):1–4, 2013.

[17] M. Batty. *The new science of cities*. The MIT Press, London, 2013.

[18] M. Batty, P. Steadman, and Y. Xie. Visualization in Spatial Modeling Michael. (0):0–18, 2004.

[19] E. Beimborn, R. Kennedy, and W. Schaefer. Inside the Blackbox: Making Transportation Models Work For Livable Communities. Technical report, 1996.

[20] D. E. Bell. Regret in Decision Making under Uncertainty. *Operations Research*, 30(5):961–981, oct 1982.

[21] K. Beven and A. Binley. The future of distributed models: Model calibration and uncertainty prediction. *Hydrological Processes*, 6(3):279–298, jul 1992.

[22] B. P. Bhatta. Discrete Choice Models – Introduction, 2016.

[23] M. Birkin and B. Wu. A Review of Microsimulation and Hybrid Agent-Based Approaches. In *Agent-Based Models of Geographical Systems*, pages 51–68. Springer Netherlands, Dordrecht, 2012.

[24] P. Boccardo and F. Giulio Tonolo. Remote sensing techniques for natural disaster impact assessment. In X. Yang and J. Li, editors, *Advances in Mapping from Remote Sensor Imagery: Techniques and Applications*, page 390. 2012.

[25] P. H. L. Bovy, M. C. J. Bliemer, and R. Van Nes. ourse CT4801: Transportation Modeling., 2006.

[26] G. E. P. Box and N. R. Draper. *Empirical model-building and response surfaces*. Wiley, 1987.

[27] BREEAM-NL. BREEAM-NL: Gebiedsontwikkeling. 2012.

[28] S. Brzev, C. Scawthorn, A. W. Charleson, L. Allen, W. Green, K. Jaiswal, and V. Silva. GEM Building Taxonomy Version 2.0. Technical report, GEM Foundation, Pavia, 2013.

[29] H. Calik, M. Labbé, and H. Yaman. p-Center Problems. In *Location Science*, pages 79–92. Springer International Publishing, Cham, 2015.

[30] Cambridge English Dictionary. Policy-making.

[31] CBS. Woningmarkt- ontwikkelingen rondom het Groningenveld. 2015.

[32] Centraal Bureau voor de Statistiek. Statistische gegevens per vierkant, 2017.

[33] Centraal Bureau voor de Statistiek. Toelichting Wijk- en Buurtkaart 2015, 2016 en 2017, 2017.

[34] Centraal Bureau voor de Statistiek. Arbeidsdeelname; regionale indeling 2015, 2018.

[35] Centraal Bureau voor de Statistiek. Banen van werknemers naar woon- en werkregio, 2018.

[36] C. Chorus. Random Regret-based discrete choice modeling: A tutorial. 2012.

[37] C. Chorus, S. van Cranenburgh, and T. Dekker. Random regret minimization for consumer choice modeling: Assessment of empirical evidence. *Journal of Business Research*, 67(11):2428–2436, nov 2014.

[38] C. G. Chorus. A New Model of Random Regret Minimization. *EJTIR Issue*, 10(2):181–196, 2010.

[39] C. G. Chorus, T. A. Arentze, and H. J. P. Timmermans. A random regret-minimization model of travel choice. *Elsevier*, 2008.

[40] C. G. C. G. Chorus. *Random regret-based discrete choice modeling : a tutorial*. Springer, 2012.

[41] A. Christodoulou, A. Kokkos, and M. Palmieri. Automated building stock data mining and classification using open source data. *Proceedings of the IASS Annual Symposium 2017 "Interfaces: architecture . engineering . science"*, 2017.

[42] P. Condon, J. Proft, J. Teed, and S. Muir. Sustainable Urban Landscapes Site Design Manual for B.C. Communities. Technical report, University of British Columbia, 2003.

[43] G. Coricelli, H. D. Critchley, M. Joffily, J. P. O'Doherty, A. Sirigu, and R. J. Dolan. Regret and its avoidance: a neuroimaging study of choice behavior. *Nature Neuroscience*, 8(9):1255–1262, sep 2005.

[44] N. Coulombel. Residential choice and household behavior : State of the Art. SustainCity Working Paper. page 70, 2010.

[45] N. Cross. *Designerly Ways of Knowing*. Springer-Verlag, London, 2006.

[46] K. de Koning, T. Filatova, and O. Bin. Bridging the Gap Between Revealed and Stated Preferences in Flood-prone Housing Markets. *Ecological Economics*, 136:1–13, jun 2017.

[47] K. de Koning, T. Filatova, and O. Bin. Improved Methods for Predicting Property Prices in Hazard Prone Dynamic Markets. *Environmental and Resource Economics*, 69(2):247–263, feb 2018.

[48] A. de Saint-Exupéry. *Wind, Sand and Stars*. 1939.

[49] W. E. Deming and F. F. Stephan. On a On a Least Squares Adjustment of a Sampled Frequency Table When the Expected Marginal Totals are Known. 1940.

[50] F. M. Dieleman. Modelling residential mobility: a review of recent trends in research. *Journal of Housing and Built Environment*, 16(1970):249–265, 2001.

[51] DSDM Consortium. *DSDM: Business Focused Development*. Addison Wesley, 2 edition, 2002.

[52] W. R. Duncan. A guide to project management body of knowledge. Technical report, 1996.

[53] D. Earnhart. Combining revealed and stated data to examine housing decisions using discrete choice analysis. *Journal of Urban Economics*, 51(1):143–169, 2002.

[54] J. M. Epstein. Why Model? oct 2008.

[55] D. Ettema, K. de Jong, H. Timmermans, and A. Bakema. Puma: Multi-Agent Modelling of Urban Systems. In *Modelling Land-Use Change*, pages 237–258. Springer Netherlands, Dordrecht, 2007.

[56] A. Evans. Uncertainty and Error. In *Agent-based models of geographical systems*, pages 309–346. Springer, 2012.

[57] T. P. Evans and S. Manson. Space, complexity, and agent-based modeling. *Environment and Planning B: Planning and Design*, 34:196–199, 2007.

[58] S. S. Fainstein. Urban planning, 2016.

[59] R. Z. Farahani, M. SteadieSeifi, and N. Asgari. Multiple criteria facility location problems: A survey, 2009.

[60] E. H. Field. Probabilistic Seismic Hazard Analysis (PSHA): A Primer. Technical report, 2005.

[61] T. Filatova, P. H. Verburg, D. C. Parker, and C. A. Stannard. Spatial agent-based models for socio-ecological systems: Challenges and prospects. *Environmental Modelling and Software*, 2013.

[62] S. García and A. Marín. Covering Location Problems. In *Location Science*, pages 93–114. Springer International Publishing, Cham, 2015.

[63] S. Geertman. Potentials for Planning Support: A Planning-Conceptual Approach. *Environment and Planning B: Planning and Design*, 33(6):863–880, dec 2006.

[64] Gemeente Den Haag. Huisvestingsvergunning aanvragen, 2018.

[65] C. Gershenson. *Complexity : 5 questions*. Automatic Press/VIP, 2008.

[66] J. Gil and J. P. Duarte. Towards an Urban Design Evaluation Framework Integrating Spatial Analysis Techniques in the Parametric Urban Design Process.

[67] J.-F. Girres and G. Touya. Quality Assessment of the French OpenStreetMap Dataset. *Transactions in GIS*, 14(4):435–459, aug 2010.

[68] V. Grimm, U. Berger, F. Bastiansen, S. Eliassen, V. Ginot, J. Giske, J. Goss-Custard, T. Grand, S. K. Heinz, G. Huse, A. Huth, J. U. Jepsen, C. Jørgensen, W. M. Mooij, B. Müller, G. Pe'er, C. Piou, S. F. Railsback, A. M. Robbins, M. M. Robbins, E. Rossmanith, N. Rüger, E. Strand, S. Souissi, R. A. Stillman, R. Vabø, U. Visser, and D. L. DeAngelis. A standard protocol for describing individual-based and agent-based models. *Ecological Modelling*, 198(1-2):115–126, sep 2006.

[69] V. Grimm and S. F. Railsback. Designing, Formulating, and Communicating Agent-Based Models. In *Agent-Based Models of Geographical Systems*, pages 361–377. Springer Netherlands, Dordrecht, 2012.

[70] J. Groeneveld, B. Müller, C. M. Buchmann, G. Dressler, C. Guo, N. Hase, F. Hoffmann, F. John, C. Klassert, T. Lauf, V. Liebelt, H. Nolzen, N. Pannicke, J. Schulze, H. Weise, and N. Schwarz. Theoretical foundations of human decision-making in agent-based land use models – A review, jan 2017.

[71] G. Y. Handler and P. B. Mirchandani. *Location on Networks: Theory and Algorithms*. MIT Press, Cambridge, Mass. ;, 1979.

[72] F. Hautbois. Bokeh vs Dash — Which is the Best Dashboard Framework for Python?, 2018.

[73] B. Head. PythonExtension: Python extension for NetLogo.

[74] D. A. Hensher, W. H. Greene, and C. G. Chorus. Random regret minimization or random utility maximization: an exploratory analysis in the context of automobile fuel choice. *Journal of Advanced Transportation*, 47(7):667–678, nov 2011.

[75] D. A. Hensher and T. U. Ton. TRESIS: A transportation, land use and environmental strategy impact simulator for urban areas. *Transportation*, 29:439–457, 2002.

[76] A. Heppenstall, N. Malleson, and A. Crooks. "Space, the Final Frontier": How Good are Agent-Based Models at Simulating Individuals and Space in Cities? *Systems*, 4(1):9, jan 2016.

[77] Q. Huang, D. C. Parker, T. Filatova, and S. Sun. A Review of Urban Residential Choice Models Using Agent-Based Modeling. *Environment and Planning B: Planning and Design*, 41(4):661–689, aug 2014.

[78] Y. Huang, A. Verbraeck, and M. Seck. Graph transformation based simulation model generation. *Journal of Simulation*, 10(4):283–309, nov 2016.

[79] N. Huynh, J. Barthélemy, and P. Perez. A heuristic combinatorial optimisation approach to synthesising a population for agent-based modelling purposes. *Jasss*, 19(4), 2016.

[80] S. Jang, S. Rasouli, and H. Timmermans. Incorporating psycho-physical mapping into random regret choice models: model specifications and empirical performance assessments. *Transportation*, 44(5):999–1019, sep 2017.

[81] M. Jaxa-Rozen and J. H. Kwakkel. PyNetLogo: Linking NetLogo with Python. *Journal of Artificial Societies and Social Simulation*, 21(2):4, 2018.

[82] M. Jelokhani-Niaraki and J. Malczewski. Decision complexity and consensus in Web-based spatial decision making: A case study of site selection problem using GIS and multicriteria analysis. *Cities*, 45:60–70, 2015.

[83] J. Jensen and M. Elle. Exploring the Use of Tools for Urban Sustainability in European Cities. *Indoor and Built Environment*, 16(3):235–247, jun 2007.

[84] R. Jordan, M. Birkin, and A. Evans. Agent-Based Modelling of Residential Mobility, Housing Choice and Regeneration. In A. Heppenstall, editor, *Agent-Based Models of Geographical Systems*, pages 511–524. Springer Science+Business, 2012.

[85] Kadaster. Objectenhandboek BAG. Technical report, 2016.

[86] Kadaster. BGT, 2018.

[87] Kadaster. BRK, 2018.

[88] D. Kahneman and A. Tversky. The psychology of preferences. *Scientific American*, 246(1):160–173, 1982.

[89] S. Karabay, E. Köse, M. Kabak, and E. Ozceylan. Mathematical Model and Stochastic Multi-Criteria Acceptability Analysis for Facility Location Problem. *Promet – Traffic & Transportation*, 28(3):245–256, jun 2016.

[90] J. Kim, F. Pagliara, and J. Preston. The Intention to Move and Residential Location Choice Behaviour. *Urban Studies*, 42(9):1621–1636, 2005.

[91] D. E. Knuth. Structured Programming with go to Statements. *ACM Computing Surveys (CSUR)*, 6(4):261–301, 1974.

[92] G. Laporte, S. Nickel, and F. da Gama. Introduction to Location Science. In *Location Science SE - 1*, pages 1–18. Springer International Publishing, Cham, 2015.

[93] B. H. Y. Lee, P. Waddel, L. Wang, and R. M. Pendyala. Reexamining the influence of work and nonwork accessibility on residential location choices with a microanalytic framework. *Environment and Planning A*, 42:913–930, 2010.

[94] J.-S. Lee, T. Filatova, A. Ligmann-Zielinska, B. Hassani-Mahmooei, F. Stonedahl, I. Lorscheid, A. Voinov, G. Polhill, Z. Sun, and D. C. Parker. The Complexities of Agent-Based Modeling Output Analysis. *Journal of Artificial Societies and Social Simulation*, 18(4), 2015.

[95] W. Li. *Modeling Household Residential Choice Using Multiple Imputation*. PhD thesis, Massachusetts Institute of Technology, 2014.

[96] G. Loomes and R. Sugden. Regret Theory: An Alternative Theory of Rational Choice Under Uncertainty. *The Economic Journal*, 92(368):805, dec 1982.

[97] C. Macal. Everything you need to know about agent-based modelling and simulation. *Journal of Simulation*, 10(27):144–156, 2016.

[98] J. Malczewski and C. Rinner. *Multicriteria Decision Analysis in Geographic Information Science*. Advances in Geographic Information Science. Springer Berlin Heidelberg, Berlin, Heidelberg, 2015.

[99] N. Malleson. Using Agent-Based Models to Simulate Crime. In *Agent-Based Models of Geographical Systems*, pages 411–434. Springer Netherlands, Dordrecht, 2012.

[100] D. Masad and J. Kazil. MESA: An Agent-Based Modeling Framework. *Proceedings of the 14th Python in Science Conference (SCIPY 2015)*, pages 53–60, 2015.

[101] D. McFadden. Conditional logit analysis of qualitative choice behavior. In P. Zarembka, editor, *Frontiers in Econometrics*, chapter 4, pages 105–142. Academic Press, New-York, 1973.

[102] Merriam-Webster. Aleatory, 2018.

[103] Merriam-Webster. Epistemic, 2018.

[104] K. Miettinen. *Nonlinear multiobjective optimization*. Kluwer Academic Publishers, Boston ;, 1999.

[105] K. Miettinen, J. Hakanen, and D. Podkopaev. Interactive Nonlinear Multiobjective Optimization Methods. In S. Greco, J. Figueira, and M. Ehrgott, editors, *Multiple Criteria Decision Analysis*, chapter 22, pages 927–976. Springer, New-York, 2016.

[106] G. A. Miller. Psychology and its history. *Psychological Review*, 63(81-97), 1956.

[107] J. Minnery and G. Searle. Toying with the City? Using the Computer Game SimCity^{TM}4 in Planning Education. *Planning Practice and Research*, 29(1):41–55, 2014.

[108] E. J. Molin. *Conjoint modeling approaches for residential group preferences*. Technische Universiteit Eindhoven, Eindhoven, 1999.

[109] P. Mooney. An Outlook for OpenStreetMap. In A. J. Jokar, A. Zipf, P. Mooney, and M. Helbich, editors, *OpenStreetMap in GIScience*, pages 319–324. Springer,, Cham, 2015.

[110] Nationaal Georegister. Ziekenhuizen in Nederland, 2013.

[111] OpenStreetMap. About OpenStreetMap, 2014.

[112] OpenStreetMap. Key:highway, 2018.

[113] J. d. D. Ortuzar and L. G. Willumsen. *Modelling transport*. Wiley, 4 edition, 2011.

[114] D. O'Sullivan and G. L. W. Perry. *Spatial Simulation: Exploring Pattern and Process*. John Wiley & Sons Inc, 2013.

[115] F. Pagliara, J. Preston, and D. Simmonds, editors. *Residential Location Choice: Models and Applications*. Springer Science & Business Media, London, 2010.

[116] S. Panagoulias, A. Laera, and R. B. J. Brinkgreve. A practical study on the induced seismicity in Groningen and the seismic response of a masonry structure. In *The 3rd International Conference on Performance Based Design (PBD-III)*, 2017.

[117] Pandas Community. Enhancing Performance — pandas 0.23.4 documentation, 2018.

[118] D. C. Parker, S. M. Manson, M. A. Janssen, M. J. Hoffmann, and P. Deadman. Multi-agent systems for the simulation of land-use and land-cover change: A review, 2003.

[119] P. Parker, R. Letcher, A. Jakeman, M. Beck, G. Harris, R. Argent, M. Hare, C. Pahl-Wostl, A. Voinov, M. Janssen, P. Sullivan, M. Scoccimarro, A. Friend, M. Sonnenshein, D. Barker, L. Matejicek, D. Odulaja, P. Deadman, K. Lim, G. Larocque, P. Tarikhi, C. Fletcher, A. Put, T. Maxwell, A. Charles, H. Breeze, N. Nakatani, S. Mudgal, W. Naito, O. Osidele, I. Eriksson, U. Kautsky, E. Kautsky, B. Naeslund, L. Kumblad, R. Park, S. Maltagliati, P. Girardin, A. Rizzoli, D. Mauriello, R. Hoch, D. Pelletier, J. Reilly, R. Olafsdottir, and S. Bin. Progress in integrated assessment and modelling. *Environmental Modelling & Software*, 17(3):209–217, jan 2002.

[120] J. Peel. How Cities: Skylines was nearly a political sim, 2017.

[121] S. Pensa, E. Masala, I. Lami, and A. Rosa. Seeing is knowing: data exploration as a support to planning. *Proceedings of the Institution of Civil Engineers - Civil Engineering*, 167(5):3–8, 2014.

[122] S. Pensa, E. Masala, and I. M. Lami. Supporting Planning Processes by the Use of Dynamic Visualisation. *Planning Support Systems for Sustainable Urban Development*, 195:451–467, 2013.

[123] C. Perpiña, J. C. Martínez-Llario, and Á. Pérez-Navarro. Multicriteria assessment in GIS environments for siting biomass plants. *Land Use Policy*, 31:326–335, mar 2013.

[124] R. G. Pontius Jr and J. Spencer. Uncertainty in extrapolations of predictive land-change models. *Environment and Planning B: Planning and Design*, 32:211–230, 2005.

[125] D. Pumain and R. Reuillon. Urban Dynamics and Simulation Models. page 123, 2017.

[126] Python Software Foundation. Debugging and Profiling — Python 3.7.0 documentation, 2018.

[127] J. Quiggin. Regret theory with general choice sets. *Journal of Risk and Uncertainty*, 8(2):153–165, mar 1994.

[128] T. H. Rashidi, J. Auld, and A. K. Mohammadian. A behavioral housing search model: Two-stage hazard-based and multinomial logit approach to choice-set formation and location selection. *Transportation Research Part A: Policy and Practice*, 46(7):1097–1107, 2012.

[129] T. H. Rashidi, A. Mohammadian, and F. S. Koppelman. Modeling interdependencies between vehicle transaction, residential relocation and job change. *Transportation*, 38(6):909–932, 2011.

[130] S. Rasouli and H. Timmermans. Applications of theories and models of choice and decision-making under conditions of uncertainty in travel behavior research. *Travel Behaviour and Society*, 1(3):79–90, sep 2014.

[131] M. Rietdijk. Catalogus basisregistraties adressen en gebouwen, 2009.

[132] Rijksoverheid. Gebruiksvoorwaarden WOZ-waardeloket, 2018.

[133] Rijksoverheid. Verblijfsobject, 2018.

[134] Rijkswaterstaat. Handleiding Nationaal Wegenbestand, 2014.

[135] Rijkswaterstaat. Handboek Kalibratie, 2015.

[136] Rijkswaterstaat. Nationaal Wegenbestand (NWB), 2017.

[137] H. W. J. Rittel and M. M. Webber. Dilemmas in a general theory of planning. *Policy Sciences*, 4(2):155–169, jun 1973.

[138] M. Roser and H. Ritchie. Natural catastrophes. *The B.E. Journal of Economic Analysis & Policy*, 8(1), 2008.

[139] Safe Software. FME Desktop, 2018.

[140] M. H. Salas-Olmedo, Y. Wang, and A. Alonso. Assessing accessibility with local coefficients for the LUTI model MARS. *Computers, Environment and Urban Systems*, 64:194–203, jul 2017.

[141] P. M. Schirmer, M. A. Van Eggermond, and K. W. Axhausen. The role of location in residential location choice models: a review of literature. *Journal of Transport and Land Use*, 7(2):3, 2014.

[142] H. S. Sevcikova, A. E. Raftery, and P. A. Waddell. Assessing uncertainty in urban simulations using Bayesian melding. *Transportation Research Part B*, 41(6):652–669, 2007.

[143] S. Sheppard. Chapter 41 Hedonic analysis of housing markets. *Handbook of Regional and Urban Economics*, 3:1595–1635, 1999.

[144] E. Silverman. *Methodological Investigations in Agent-Based Modelling - With Applications for the Social Sciences*. SpringerOpen, Cham, 2014.

[145] H. A. Simon. Theories of bounded rationality. In C. B. McGuire and R. Radner, editors, *Decision and organization*, chapter 8, pages 161–176. North-Holland Publishing Company, 1972.

[146] H. A. Simon. *The sciences of the artificial*. MIT Press, 3 edition, 1997.

[147] R. C. Spear. The application of Kolmogorov-Rényi statistics to problems of parameter uncertainty in systems design. *International Journal of Control*, 11(5):771–778, 1970.

[148] Stichting Lisa. Gratis data, 2018.

[149] Sunlight Foundation. Ten Principles For Opening Up Government Information, 2010.

[150] Synthicity. UrbanCanvas — UrbanSim, 2017.

[151] R. G. Tabak, E. C. Khoong, D. A. Chambers, and R. C. Brownson. Bridging Research and Practice: Models for Dissemination and Implementation Research. *American Journal of Preventive Medicine*, 43(3):337–350, sep 2012.

[152] R. Tanton, P. Williamson, and A. Harding. Comparing Two Methods of Reweighting a Survey File to Small Area Data. *Microsimulation association international journal of microsimulation*, 7(1):76–99, 2014.

[153] The Scipy community. numpy.histogram — NumPy v1.13 Manual, 2017.

[154] J. C. Thiele. *Towards Rigorous Agent-Based Modelling Linking, Extending, and Using Existing Software Platforms*. Phd thesis, Niedersächsische Staats-und Universitätsbibliothek Göttingen, 2014.

[155] W. Ulrich. C. West Churchman– 75 Years. *System Practice*, 1(4):341–350, 1988.

[156] U.S. Green Building Council. LEED, 2018.

[157] S. van Cranenburgh. Latent class models, 2015.

[158] S. van Cranenburgh, C. A. Guevara, and C. G. Chorus. New insights on random regret minimization models. *Transportation Research Part A: Policy and Practice*, 2015.

[159] H. van Delden, R. Seppelt, R. White, and A. Jakeman. A methodology for the design and development of integrated models for policy support. *Environmental Modelling & Software*, 26(3):266–279, mar 2011.

[160] K. van Thienen-Visser and J. N. Breunese. Induced seismicity of the Groningen gas field: History and recent developments. *The Leading Edge*, 34(6):664–671, 2015.

[161] B. van Wee. Accessible accessibility research challenges. *Journal of Transport Geography*, 51:9–16, feb 2016.

[162] M. Velasquez and P. T. Hester. An Analysis of Multi - Criteria Decision Making Methods. *International Journal of Operations Research*, 10(2):56–66, 2013.

[163] G. Vonk, S. Geertman, and P. Schot. Bottlenecks Blocking Widespread Usage of Planning Support Systems. *Environment and Planning A*, 37(5):909–924, may 2005.

[164] P. Waddell. UrbanSim: Modeling Urban Development for Land Use, Transportation and Environmental Planning. *Journal of the American Planning Association*, 68(3):297, 2002.

[165] P. Waddell. Integrated Land Use and Transportation Planning and Modelling: Addressing Challenges in Research and Practice. *Transport Reviews*, 31(2):209–229, mar 2011.

[166] P. Waddell, G. Boeing, M. Gardner, and E. Porter. An Integrated Pipeline Architecture for Modeling Urban Land Use, Travel Demand, and Traffic Assignment, feb 2018.

[167] P. Waddell and G. F. Ulfarsson. Introduction to Urban Simulation: Design and Development of Operational Models, 2004.

[168] P. Wagner and M. Wegener. Urban Land Use, Transport and Environment Models Experiences with an Integrated Microscopic Approach. *DisP-the planning review*, 43(170):45–56, 2007.

[169] M. Wegener. Land-use transport interaction models. In *Handbook of Regional Science*, pages 741–758. Springer, Heidelberg, 2014.

[170] M. Wegener and F. Fuerst. Land-Use Transport Interaction. *SSRN Electronic Journal*, jan 2004.

[171] Wikipedia Contributors. Floyd–Warshall algorithm, 2018.

[172] U. Wilensky. NetLogo User Manual, 2018.

[173] A. Wilson. *The Science of Cities and Regions*. Springer Netherlands, Dordrecht, 2012.

[174] G. K. Wong. A Conceptual Model of the Household's Housing Decision-Making Process: The Economic Perspective. *Review of Urban and Regional Development Studies*, 14(3):217–234, nov 2002.

[175] E. K. Zavadskas, Z. Turskis, and S. Kildienė. State of art surveys of overviews on MCDM/MADM methods. *Technological and Economic Development of Economy*, 20(1):165–179, jan 2014.

[176] M. Zeelenberg. Anticipated regret , expected feedback and behavioral decision-making Anticipated Regret , Expected Feedback and Behavioral Decision Making. *Journal of behavioral decision making*, 106(September 1998):93–106, 1999.

[177] J. Zhang, B. Yu, and M. Chikaraishi. Interdependences between household residential and car ownership behavior: A life history analysis. *Journal of Transport Geography*, 34:165–174, 2014.

[178] B. Zondag, M. de Bok, K. T. Geurs, and E. Molenwijk. Accessibility modeling and evaluation: The TIGRIS XL land-use and transport interaction model for the Netherlands. *Computers, Environment and Urban Systems*, 49:115–125, jan 2015.

[179] B. Zondag and G. de Jong. The development of the TIGRIS XL model: a bottom-up approach to transport, land-use and the economy. *Research in Transportation Economics*, 31(1):55–62, 2013.

Appendices

A IMPLEMENTATION

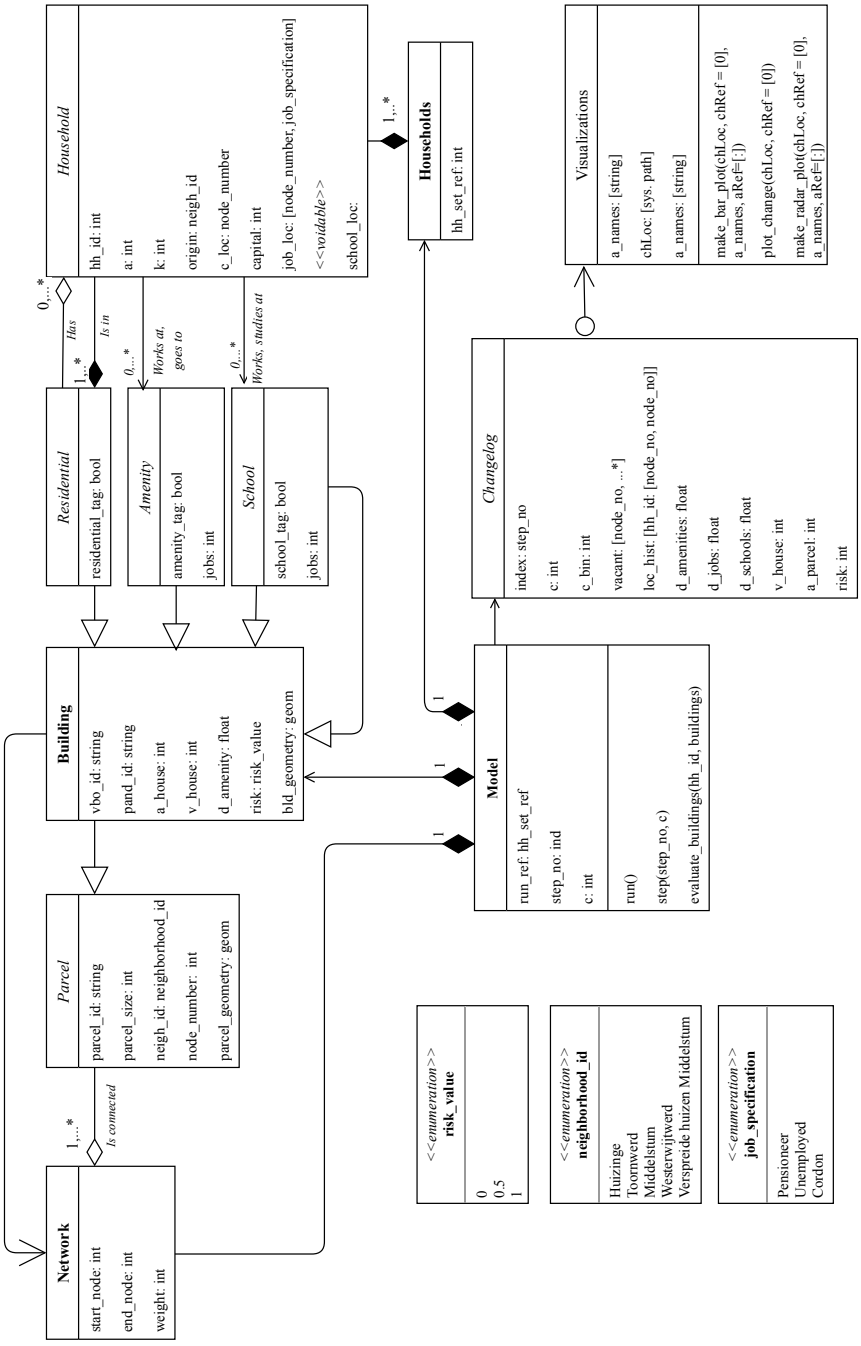


Figure A.1: Model classes, their properties and most important methods

A.1 (PSEUDO-)CODE

All of the code was written using Python 3.14, making use o5f the libraries numpy, pandas, geopandas, shapely

A.1.1 Problem definition

Location permutation number:

```python
import numpy as np

k = 6000
p = 1260
total = 1261
for n in range(1262, 7261):
    total = total * n
permutations = []
N = k
for n in range(N):
    if n == 0:
        continue
    for m in range(n):
        permutations.append((k-m)*(p+n-m-1))
sum_perm =sum(permutations)
all_scenarios = (total-sum_perm)*2+3*(sum_perm)
print (all_scenarios)
```

A.1.2 Network

Algorithm A.1: Distance matrix construction [171]

Data: Distance matrix D
Result: Shortest path matrix D (2x2)

```
foreach edge (u,v) do
    dist[u][v] ← w(u,v);
foreach vertex v do
    dist[v][v] ← 0;
for k from 1 to V do
    for i from 1 to V do
        for j from 1 to V do
            if dist[i][j] > dist[i][k] + dist[k][j] then
                dist[i][j] ← dist[i][k] + dist[k][j];
```

Algorithm A.2: Distance to jobs or schools calculation, based on the distance matrix

Data: Distance matrix D <array>, current location c_loc <str>, job or school locations ref_loc <list>
Result: Aggregate distance to jobs d_ref<int>

```
d_ref ← 0;
for i from 0 to length(ref_loc) do
    j ← ref_loc[i];
    if j != 'cordon' and j != 'pensioneer' and j != 'unemployed' then
        d_ref ← d_ref + D[c_loc, ref_loc[i]];
```

Algorithm A.3: N closest amenity identification and distance calculation based on the distance matrix

Data: Distance matrix D <array>, current location c_loc <str>, buildings table B, number of amenities N <int>
Result: Aggregate distance to 3 closest amenities <int>

```
homes ← B['residential_flag'==True];
amenity_nodes ← B[B['amenity_flag'==True], 'node_number'];
D ← D.filter(amenity_nodes);
for i from 0 to len(homes) do
    h_node ← B[i, 'node_number'];
    // finding distances to N closest amenities, given a
       location on a network
    B[i, 'd_amenities'] ← D[h_node].nsmalles(N);
```

A.2 CODE VERIFICATION

In the code we have two types of errors, that are not native to Python or its libraries: *PopulationError* and *ModelError*. As the name suggests, the first type of error only occurs, when some property, relating to the population (i.e. households) does not have a valid input. What we define as valid in the data-preprocessing and synthesizing steps is corresponding to the statistics between datasets on zonal and aggregate scale. For instance, during household synthesizing amount of active, working population in the case study area and coming from outside should be equal to the number workplaces. But this also applies to the simulation as well: the average population per household should be the same before and after each model steps. The second error, ModelError, occurs if the inputs to one of the functions in the code are not according to specifications. For example, if a household has multiple current residence locations or a house has more than one household residing in it.

A.3 COMPUTER SPECIFICATIONS

	Major runs and time estimations	High-end processor	Secondary-device
Processor	Intel(R) Core(TM) i7-6600U CPU @ 2.60GHz	Intel Xeon E5-1620 v3 @ 3.5GHz	Intel(R) Core(TM) i7-6700HQ CPU @ 2.60GHz
RAM	16 GB	32 GB	6 GB

B ADDITIONAL RESULTS

B.1 BASE RUN

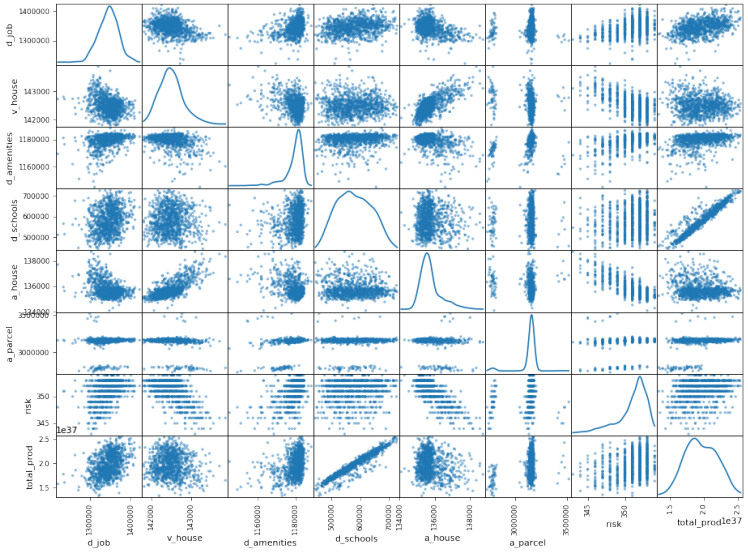


Figure B.1: Pairwise attribute scatter plots for the base run with 1000 household sets

B.2 DYNAMIC RISK

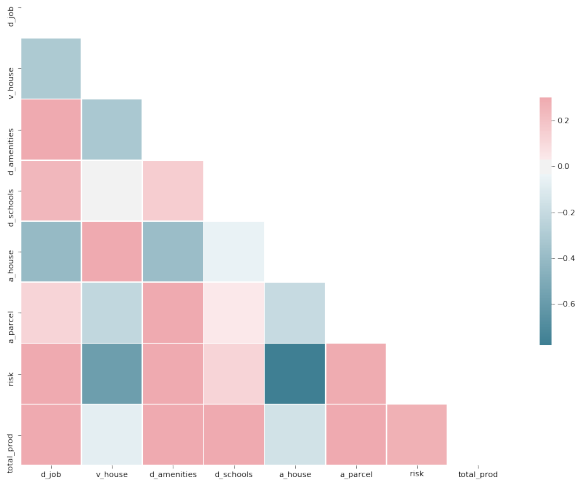


Figure B.2: Pairwise attribute correlation heat-map for the base run with 1000 household sets

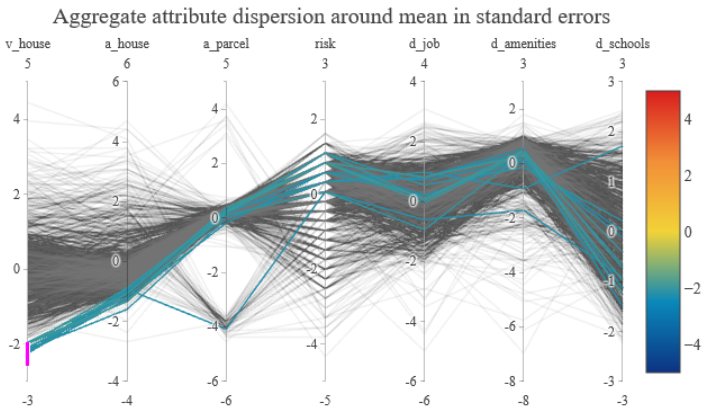


Figure B.3: Parallel coordinate plot for all attributes (in standard errors) for the base run with 1000 household sets. Colored by the value of the attribute v_house. Filter is to show the distributions of other attribute values

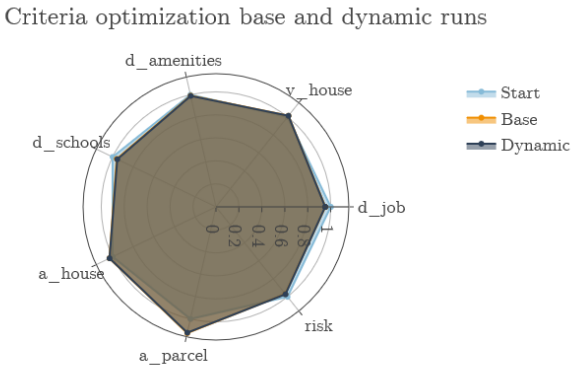


Figure B.4: Aggregate criteria optimization in relation to the starting condition for base and dynamic risk runs ($\beta = 1$) for 100 household sets, Middelstum scale

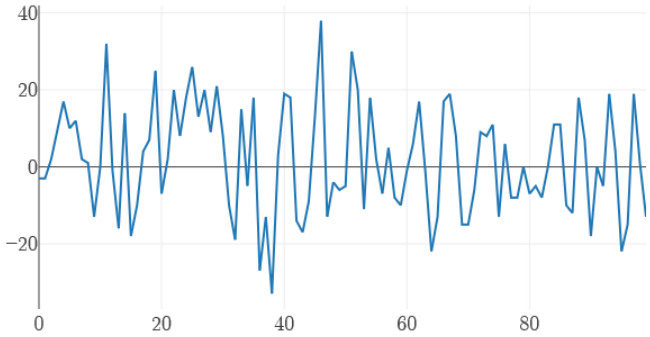


Figure B.5: Differences between total number of upgrades per each household set (1 to 100), given $\beta_{risk} = 1$ or 4

COLOPHON

This document was typeset using LaTeX. The document layout was generated using the `arsclassica` package by Lorenzo Pantieri, which is an adaption of the original `classicthesis` package from André Miede.

All images, unless specified otherwise have been generated by the author. Plots were generated primarily using Plotly library for Python. For vector images Inkscape and Illustrator were used.

ISBN 978–94–6186–989–0