2023-09-01

Multi-agent Spatiotemporal Simulation of Autonomous Vehicle Fleet Operation

ZHANG, ZONGHAO

UNIVERSITY OF CALGARY

Multi-agent Spatiotemporal Simulation of Autonomous Vehicle Fleet Operation

by

Zonghao Zhang

A THESIS

SUBMITTED TO THE FACULTY OF GRADUATE STUDIES

IN PARTIAL FULFILMENT OF THE REQUIREMENTS FOR THE

DEGREE OF MASTER OF SCIENCE

GRADUATE PROGRAM IN GEOMATICS ENGINEERING

CALGARY, ALBERTA

SEPTEMBER, 2023

ABSTRACT

Autonomous vehicle fleets, consisting of self-driving vehicles, are at the forefront of transportation innovation. The appearance of autonomous vehicles (AVs) provides a new solution for traffic problems and a new market for transportation network companies such as DiDi and Uber. Conducting simulations in the present is indeed crucial to prepare for the eventual operation of autonomous vehicles, as their widespread adoption is expected to occur in the near future. This research adopts an Agent-Based Modelling (ABM) approach to understand and optimize the performance of autonomous vehicle systems. Moreover, Geographic Information System (GIS) technology also plays a crucial role in enhancing the effectiveness and accuracy of the simulation process. GIS enables the representation and manipulation of geospatial data, such as road networks, land-use patterns, and population distribution. The combination of ABM and GIS allows for the incorporation of real-world geographic data, providing a realistic and geographically accurate environment for the agents in the virtual environment. In this thesis, the multi-agent spatiotemporal simulation is conducted by the GAMA platform. The model simulates the behaviour and interactions of individual agents, which are fleet agents and commuters, to observe the emergent behaviour of the entire system. Within the experiment, different scenarios are considered for both people and fleets to explore a range of approaches and strategies. These scenarios aim to evaluate the effectiveness of various approaches in meeting dynamic commute needs and optimizing fleet operations. By simulating these different scenarios and analyzing their outcomes, the study aims to provide insights into the improvement of fleet size and deployment in autonomous vehicle systems. The ultimate goal is to identify effective strategies that lead to optimized fleet size in

different scenarios, reduced idling time and emission, improved traffic management, and overall more efficient and sustainable autonomous vehicle systems.

PREFACE

This dissertation is an original, unpublished, independent work by the author, Zonghao Zhang.

ACKNOWLEDGEMENTS

I would like to thank the following persons for their invaluable counsel, exceptional leadership, and unwavering faith in my work.

First of all, I am deeply grateful to my supervisor, Dr. Emmanuel Stefanakis, Head of the Department of Geomatics Engineering at the University of Calgary. His guidance, expertise, and unwavering support have been instrumental in shaping this research project and my overall academic development.

I extend my heartfelt thanks to my undergraduate mentor, Dr. John Olusegun Ogundare, whose dedication and passion for research inspired me and instilled in me a sense of perseverance and curiosity. I am truly grateful for the valuable lessons and knowledge I gained under his mentorship.

I would like to express my deepest appreciation to my family for their unconditional love, unwavering encouragement, and constant support throughout my academic journey. I am deeply indebted to my parents for their tremendous love and encouragement. Their belief in me and their financial assistance have played a crucial role in enabling me to pursue my studies and research aspirations. I would also like to extend my sincere thanks to my cousins. Thank you for always being there to lend a listening ear, offer words of encouragement, and celebrate my achievements. Your belief in me and your constant encouragement have played a significant role in my success.

I am indebted to Dr. Mingke Li for her guidance, wisdom, and companionship throughout this research endeavor. Her expertise, valuable insights, and continuous encouragement have been pivotal in the successful completion of this work.

Additionally, I would also like to extend my gratitude to Dr. Yaser Sadeghi for the valuable discussions, ideas, and feedback provided before the start of this academic journey. His expertise and constructive input have greatly contributed to the quality and depth of my academic journey.

My heartfelt thanks go to my friends, especially Mr. Hao Yan, Mr. Ziao Tan, Mr. Hao Luo, Dr. Juqing Liu, Dr. GuanTian Yang, and Dr. Jianv Huang. Your unwavering support, encouragement, and friendship have been invaluable throughout my academic journey. The countless discussions, brainstorming sessions, and shared experiences have not only enriched my research work but have also shaped me as an individual. I am grateful for the long-lasting friendship we have cultivated, characterized by mutual support, respect, and shared aspirations.

I am grateful to my colleagues and the entire research team for their support, collaboration, and stimulating discussions. Their presence and contributions have enriched my research experience and made this journey more enjoyable.

Finally, I would like to extend my heartfelt thanks to myself, acknowledging the remarkable personal growth and journey I have undertaken. I started as a student with below-average grades in high school in Nanjing, China and faced the challenge of not being fluent in English when I arrived in Canada at the age of 19. I express gratitude to myself for never losing sight of my dreams and for remaining persistent in my pursuit of higher education. The journey I have embarked upon has shaped me into a more relentless individual, instilling within me a sense of determination, self-belief, and a commitment to personal growth. As I reflect on my achievements, I am proud of the progress I have made. I appreciate the dedication and hard work that I have invested in my education, and I am grateful for the continuous efforts I have made to improve myself and my life.

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

LIST OF ABBREVIATIONS

ABM – Agent-Based Model

ABMS – Agent-Based Modeling and Simulation

GIS – Geographic Information System

AV – Autonomous Vehicle

IDE – Integrated Development Environment

HPC – High Performance Computing

QGIS – Quantum Geographic Information System

TTS – Transportation Tomorrow Survey

GAML – GAma Modeling Language

OSM – Open Street Map

CHAPTER 1: INTRODUCTION

1.1 BACKGROUND

Transportation systems play a vital role in modern cities, which service millions of individuals each day. The operation of the transportation system conducts all human activities. However, urban transportation systems are facing many complex challenges nowadays, such as traffic congestion, limited parking availability, loss of public space, negative environmental impacts, and issues of transport equity. These challenges impact cities on a global scale that has never happened before, adversely affecting city development and human life. Furthermore, it is important to note that these challenges are not independent entities; instead, they interact and influence one another. The lack of parking availability and traffic congestion are two of the most pressing issues that need to be mitigated promptly, which are prevalent in areas with high population densities with limited infrastructure. Congestion can result in an increased rate of accidents, reduced productivity, and prolonged commute times. In addition, the congestion increases the vehicle's idling time, which can result in increased air pollution and greenhouse gas emissions. Another common problem in the current urban transportation system is the lack of parking spaces, which causes more idling time and raises air pollution. Indeed, the shortage of parking places and traffic congestion are interconnected problems that mutually influence and exacerbate each other. The limited availability of parking spaces leads to a constant search for parking, resulting in increased traffic volume and congestion. This, in turn, worsens the congestion problem by reducing the flow of traffic and increasing travel times. The continuous circulation of vehicles searching for parking spaces creates a vicious cycle, further intensifying the congestion and parking shortage. Therefore, addressing these issues requires a comprehensive approach that considers both the availability of parking infrastructure and effective traffic management strategies to break this cycle and improve the overall transportation system's efficiency.

The usage of autonomous vehicles should be encouraged to address the issues of urban transportation. As stated in (Maciejewski & Bischoff, 2018), autonomous vehicle fleet services have the potential to significantly reduce the number of vehicles needed and address the challenges associated with extensive parking spaces. By promoting the use of autonomous vehicles, traffic congestion can be reduced because autonomous vehicles can communicate with one another and enhance the efficiency of routes. People who use autonomous vehicles as their commute choice do not need to worry about finding a parking spot. In addition, autonomous vehicles can also help reduce the negative impacts of climate change by using hybrid or electric powertrains. Since the transportation system is a complex system, and many factors are interconnected, the deployment of the autonomous vehicle fleet should be tested through the transportation planning process.

Transportation planning plays a crucial role in shaping the future of urban mobility and ensuring efficient and sustainable transportation systems. Informed decision-making is a key component of effective transportation planning, as it relies on accurate data, comprehensive analysis, and a deep understanding of the complex dynamics of transportation networks. In order to test and make informed decisions for the deployment of the autonomous vehicle fleet accurately, the simulation of the deployment is required. It allows transportation planners and decision-makers to create virtual environments that mimic real-world conditions and dynamics. Simulations offer the ability to input diverse parameters like fleet size, demand patterns, operational strategies, and infrastructure configurations, which allows for the generation of insights and the evaluation of performance across different deployment scenarios.

1.2 RESEARCH AND OBJECTIVES

The emergence of autonomous vehicles (AVs) has the potential to provide a new solution for traffic problems and create a new market for transportation network companies. By addressing the root cause of bad driving behaviour, AVs can potentially reduce the required total vehicle fleet size (Boesch et al., 2016). Moreover, AVs can provide increased accessibility for people who cannot drive, including disabled individuals, older adults, and unlicensed individuals (Jing et al., 2020). According to a study by (Bösch et al., 2018), the shared autonomous vehicle fleet has drastically reduced costs per passenger-kilometer compared to the conventional taxi fleet. These advantages increase accessibility and lower cost, which make autonomous vehicle fleets a viable alternative to current commute options. While widespread adoption of AVs may be some ways off, simulations need to be carried out to prepare for the future operation of autonomous vehicle fleets, which can help reduce operating costs and test different operational strategies to meet dynamic commuting needs. Optimizing the fleet size and deployment based on demand changes is one of the solutions that can significantly reduce operational costs and minimize the idling time of individual vehicles. With all cars being autonomous, idle vehicles can be deactivated and removed from the system until needed again. Such an operation strategy can directly reduce emissions and traffic congestion. By analyzing the dynamics of demands on various spatial and temporal scales, the model can demonstrate the autonomous vehicle fleet's size variation trends. This simulation can help operators better allocate and manage resources to meet the changing needs of commuters, ultimately leading to a more sustainable and efficient transportation system.

The use of agent-based modelling (ABM) is a valuable technique in generating a bottom-up simulation of a system to understand better the output of interactions between individual agents and the dynamic environment. ABM is composed of three key elements, as Figure 1 illustrates: agents, agent relationships and methods of interaction, and the agent's environment (Macal & North, n.d.). Agents in the ABM framework represent the micro-level entities and engage in interactions based on their relationships and prescribed methods. The collective behaviour and relationships of these individual agents give rise to macro-level outcomes within the environment. In the context of transportation systems, the intricate interplay of spatial and temporal parameters, along with evolving commuter demands, necessitates a simulation approach that extends beyond the limitations of mathematical equations alone. The ABM offers a suitable methodology to understand the underlying mechanisms governing these complex interactions and behaviours, allowing for a more comprehensive analysis of transportation scenarios. By capturing the emergent behaviour resulting from agent interactions, the ABM enables researchers to study the dynamics of transportation systems in a realistic and informative manner.

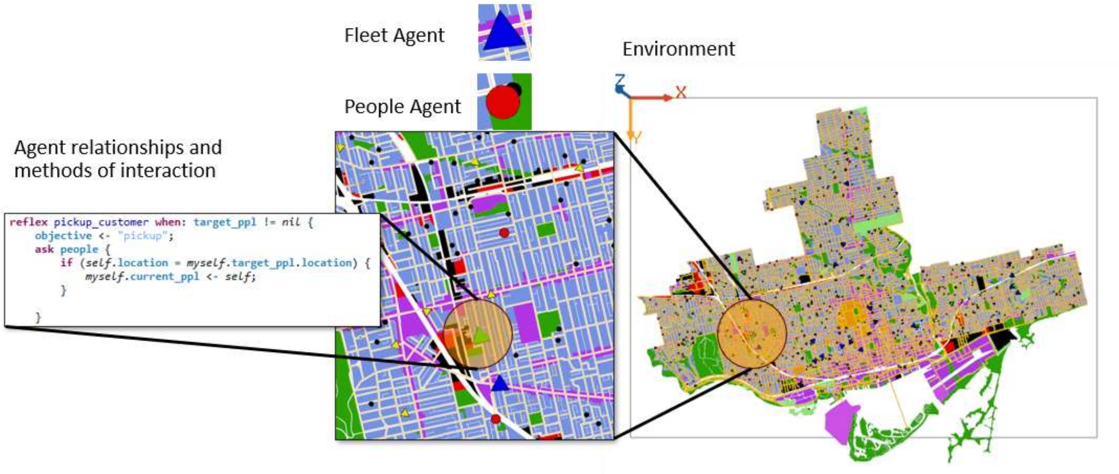

Figure 1 GAMA Platform contexts: the agents, agent relationships and method of interactions and the environment

Figure 2 showcases the utilization of an agent-based model (ABM) to simulate the operation of a taxi fleet at a city scale. The simulation incorporates essential modelling parameters, including car speed, service area, and maximum taxi capacity, within a framework of a road network and buildings. The ABM consists of two agent types: taxi cars and people. Taxi cars are responsible for meeting the demands generated by the people agents. As taxi agents navigate the road network to transport people to their desired destinations, they interact with the simulated world and adhere to constraints such as speed limits. This simulation offers valuable insights into the dynamics of taxi fleet operations within a city, enabling experimentation with different parameters to optimize efficiency and alleviate congestion.

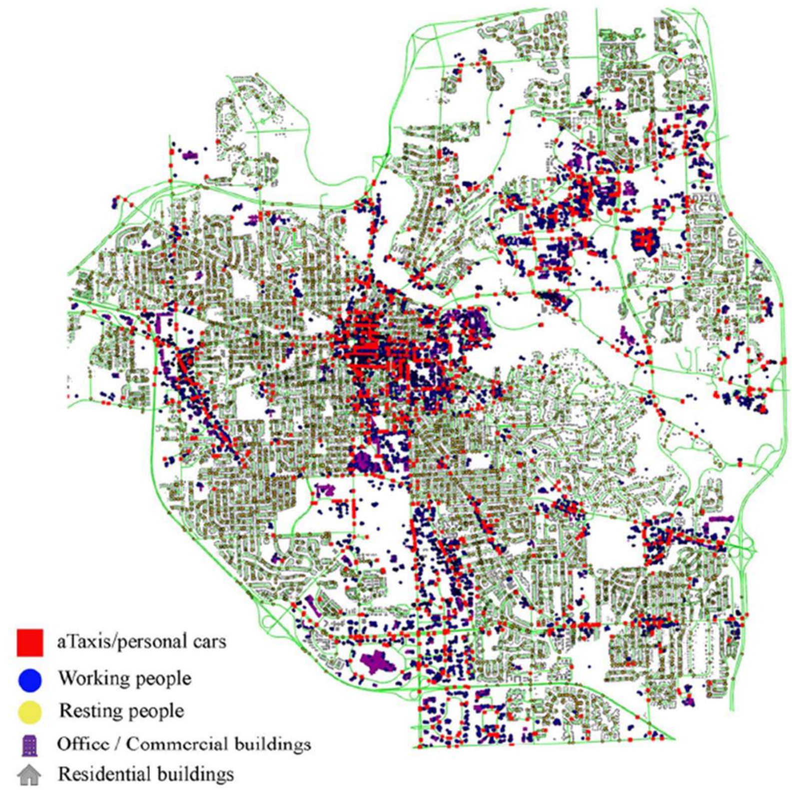

Figure 2 City of Ann Arbor Autonomous Taxi Simulation: the road network, people and vehicle

Source: Adapted from (Lu et al., 2018)

The agent-based model (ABM) was chosen as the preferred methodology for this research due to its modular and flexible nature, which allows for testing various scenarios and objectives. This thesis aims to assess the effectiveness of different fleet sizes and deployment strategies in reducing operational costs, minimizing idling time, and mitigating emissions and traffic congestion. The simulation incorporates a virtual environment, people agents, and vehicle agents. Once the virtual world is constructed and the foundational parameters are established, different types of fleet operations and demand generation can be simulated and evaluated using the same environment settings. This can be achieved by adjusting the parameters of the people and vehicle agents within the environment. Furthermore, the simulation can be assessed by keeping the agent settings constant while varying the world parameters, such as the waiting time constraints for vehicles during different time periods. The ABM approach excels in capturing spatiotemporal characteristics, enabling the generation of a diverse population of agents and simulating variations in travel demand among various regions in different times of the day.

1.3 THESIS ORGANIZATION

The remaining sections of this thesis are organized into four main chapters. Chapter two provides a comprehensive literature review, examining the topics of agent-based modelling, geographic information systems, and autonomous vehicles. It synthesizes existing research and identifies the key insights and gaps in the literature.

Chapter three delves into the construction of the agent-based model, presenting the proposed multi-agent model in detail. It outlines the platforms and data sources utilized in this thesis, highlighting their specific roles and the Overview, Design, Details (ODD) protocol. Additionally, this chapter explores the architecture of the model, discussing the attributes, species, and behaviours incorporated into the simulation.

Chapter four centers around the case study and experiments conducted using the developed model. It showcases processes of data cleaning and editing and the results obtained from various experiments designed to test and evaluate the implementation of the fleet under different operational strategies. The chapter provides an analysis and interpretation of the outcomes, offering valuable insights into the performance and effectiveness of the simulated scenarios.

Lastly, chapter five concludes the thesis by summarizing the key findings and contributions of the research. It also includes a discussion of the limitations encountered during the study and outlines potential avenues for future research and development in this field. This final chapter serves as a reflection on the research journey and offers recommendations for further exploration and improvement.

CHAPTER 2: LITERATURE REVIEW

2.1 AGENT-BASED MODEL

Agent-based modelling (ABM) has emerged as a powerful tool for understanding and simulating complex systems, including social, economic, and ecological phenomena. Unlike traditional approaches that rely on aggregated data, ABM focuses on individual entities (agents) and their interactions with each other and the environment to represent real-world dynamics. By capturing the behaviour of autonomous agents and the environment, ABM offers unique insights into system-level patterns and emergent phenomena. As stated in (Macal & North, n.d.), agent-based modelling is motivated by its capacity to capture emergence, where complex behaviours arise from simple rules and local interactions among agents. Even in basic models with deterministic rules and limited information, agents can self-organize and exhibit behaviour that was not explicitly programmed.

The typical structure of Agent-Based Models (ABMs) has three key elements:

- Agents: The individual unit in the environment with defined attributes and behaviours.
- Agent relationships and methods of interaction: An underlying topology of connectedness defines how and with whom agents interact.
- The environment: The space that all agents live in. This is the place where all interaction happens. The agents interact with the environment and other agents.

In agent-based modelling (ABM), an agent refers to an individual entity that possesses certain characteristics, behaviours, and decision-making capabilities. An agent can represent any creature with any autonomy. Individuals, structures, vehicles, land parcels, water droplets, and insects are among the examples (Crooks, 2018). During the simulation, a group of heterogeneous agents are placed in their own space within an artificial world, which forms the environment. The agents are able to interact with other agents or the environment. Through their interactions, the agents can share knowledge or concepts, which may create new ideas or knowledge. With new ideas or knowledge, the agents may make new decisions and try to achieve their goals through other approaches.

The core feature of ABM is attempting to replicate the behaviour of individuals within a system by defining agents, their attributes, and rules for interaction and placing them in realistic environments. Agent-based modelling enables the study of their interactions with each other and their surroundings (Crooks, 2018). Agents, as flexible problem solvers, operate in dynamic environments with limited control and observability. Therefore, interactions need to be handled flexibly, allowing agents to make runtime decisions and initiate unforeseen interactions. In many cases, agents act to accomplish goals within an organizational setting, either on behalf of specific people or as part of a larger effort to solve problems. This context defines the relationship between agents and influences their behaviour, such as whether they are peers collaborating in a team or if one agent serves as the manager of others (Jennings, 2000). Additionally, the use of agent-based modelling allows for the representation of individual behaviours in a spatial and temporal context. In the study of (Heppenstall et al., 2006), individual petrol stations were represented as agents in a Java-based model. These agents were equipped with information about their initial starting prices, production costs, and the prices of neighbouring stations within a defined radius. The agents had the ability to observe the prices of neighbouring stations and adjust their own prices based on a set of rules. The model ran iteratively, with each iteration representing a day, and the stations adjusting their prices based on their competitors' prices.

2.1.1 ODD PROTOCOL AND BDI MODEL

Agent-based models have been established on solid methodological foundations. However, the flexibility they offer to researchers in terms of model design has sometimes led to a lack of standardization in design, analysis, and presentation (Leombruni et al., n.d.). Furthermore, the implementation details of agent-based simulations are often insufficiently documented, making replication difficult or even impossible (Leombruni et al., n.d.). As mentioned by (Grimm et al., 2006), compared with traditional analytical models, Agent-Based Models (ABMs) are more challenging to understand, analyze, and convey because of the complexity of the structure. To enhance transparency and reproducibility in ABM, the ODD (Overview, Design concepts, Details) protocol should be considered. ODD provides a structured framework for documenting the essential aspects of an ABM study. The standardized protocol, such as ODD, for describing Agent-Based Models (ABMs) would simplify reading and understanding. Meanwhile, the problem of lengthy verbal descriptions, which hinder information extraction for understanding and implementing the model, can be addressed (Grimm et al., 2006). The structure of ODD protocol is illustrated in Table 1 below:

Overview	Purpose
	Entities, state variables and scales
	Process overview and scheduling
Design concepts	Design concepts
Details	Initialization
	Input
	Submodels

Table 1 The Three Major Blocks with Seven Elements of the ODD Protocol

The overview section includes the purpose, state variables and scales, and process overview and scheduling (Grimm et al., 2006). The initial step in developing a model is to state its purpose clearly. This is crucial as it enables readers to comprehend why certain aspects of reality are included while others are disregarded. Declarations of entities, state variables and scales include low-level entities and high-level entities. The term ‘low-level entities’ refers to individual agents in the model, and the state variables of individual agents include basic attributes such as age, sex, social rank, and location. The high-level entities, for example, the population of one species or a community consisting of populations (Grimm et al., 2006). Additionally, the scale of the model should be stated. It includes indicating the length of time steps and the overall time horizon, the size of habitat cells (if the model is grid-based), and the extent of the model world (if the model is spatially explicit). The rationale behind selecting these scales should be briefly explained since the choice of scale significantly influences the design of the entire model (Grimm et al., 2006). Finally, the element named ‘Process overview and scheduling’ answers the following questions:

- Who (i.e., what entity) does what, and in what order?
- When are state variables updated?
- How is time modelled as discrete steps or a continuum over which continuous processes and discrete events can occur?

The design concepts section outlines the key components, including emergence, adaptation, fitness, prediction, sensing, interaction, stochasticity, collectives, and observations (Grimm et al., 2006):

- Emergence: Which system-level phenomena actually result from individual attributes, and which ones are merely imposed?

- Adaptation: What adaptable characteristics do the model subjects possess that, either directly or indirectly, can increase their prospective fitness in response to changes in either their own self or their environment?
- Fitness: Does the model explicitly model fitness-seeking behaviour, or is it modelled implicitly? If it is explicitly modelled, how do individuals calculate their fitness, or in other words, what is the measure of fitness used?
- Prediction: How can agents anticipate the future circumstances they will encounter when calculating the effects of their decisions?
- Sensing: What internal and external state factors are agents supposed to "sense" or "know" and take into account while making adaptive decisions?
- Interaction: What types of interactions between agents are presupposed?
- Stochasticity: Is stochasticity incorporated into the model?
- Collectives: Are agents organized into a collective, such as a social group?
- Observations: How are ABM data gathered for use in testing, comprehension, and analysis?

The details section provides more in-depth information on initialization, data input and submodels. The initialization focuses on the creation of the environment and individuals at the beginning of a simulation run in an agent-based model (ABM). It addresses questions about the initial values of state variables, whether the initialization is consistent or varied across simulations, and whether the initial values were chosen arbitrarily or based on data. The environmental conditions are considered as the "inputs" that affect specific state variables in the model. It is essential for readers to know the details of the input data, which answers the questions about how they were generated and how they can be obtained or reproduced. In the submodels section, all the submodels that represent the processes mentioned in the "Process overview and scales" are presented and explained in a comprehensive manner. This includes providing detailed information about the parameterization of the model and specifying how the various components and interactions are modelled and calibrated.

The Belief-Desire-Intention (BDI) software model is a cognitive framework used to describe and develop intelligent agents, which is a popular approach used in ABM. It is based on the philosophical concept of beliefs, desires, and intentions as fundamental elements of agent behaviour. In the BDI model, agents possess beliefs about their environment, desires or goals they wish to achieve and intentions which represent their planned courses of action. In other words, beliefs are typically stored in a database system, representing the agent's understanding of the world. Desires are expressed as goals that the agent wants to achieve. Plan rule templates are stored in a plan library and are instantiated at runtime based on the agent's beliefs and goals. The intentions of the agents are represented by the plan instances (Singh et al., 2016). In 2012, (Taillandier et al., n.d.) proposed a new BDI architecture based on belief theory. When an agent receives new information either through its own perception or from another agent's message, it updates its belief base accordingly. If the agent does not currently have any chosen intention, it evaluates each plan in its plan base based on its desires and beliefs. Through a multicriteria decision-making process, the agent selects a plan that best aligns with its goals and beliefs and adds it to its intention base. The agent then continuously selects actions from the chosen plan based on its context, determined by its beliefs and desires. At each simulation step, the plan can be deleted or updated through a plan execution control process, allowing the agent to adapt its actions based on the changing circumstances (Taillandier et al., n.d.). However, according to (Taillandier et al., 2017), this architecture was closely tied to its application context, which had limitations in representing the agent's beliefs formally and lacked the capability to handle complex plans with sub-objectives. A simple BDI architecture was proposed by (Caillou et al., 2017) in 2017, which is easy to understand and applicable to different research fields. With a reactive agent model, the agent's behaviour is determined by simple reflexes which react to immediate events. A cognitive model considers the agent's desires and goals. The agent has a cognitive process that determines what it wants to achieve and how to achieve it. For example, the agent may have the desire to find fires and extinguish them. It plans its actions based on these desires, such as patrolling to find fires, extinguishing the fire, and refilling water when needed (Caillou et al., 2017). To summarize, the BDI architecture enables agents to make autonomous decisions based on their internal states and external stimuli. By incorporating BDI into ABM, researchers can capture the cognitive aspects of agent behaviour and simulate realistic decision-making processes.

2.1.2 PLATFORMS AND APPLICATIONS

Nowadays, Agent-based simulations are widely used to carry out research for complex systems. Several software platforms facilitate ABM development, each with its own strengths and limitations. The table below provides a comparison of various Agent-Based Modeling (ABM) platforms and software. NetLogo offers a user-friendly interface and a large library of pre-built models, while AnyLogic supports multiple modelling paradigms and advanced visualization. Repast and GAMA provide flexible and extensible frameworks with high-performance computing capabilities. Mason and SWARM focus on lightweight and efficient ABM frameworks with support for spatial modelling. FAME and MESA offer agent-based modelling frameworks with modular designs and support for Java and Python programming. Table 2 also highlights the model scale and GIS capability of each platform. However, it's important to consider the specific requirements of a modelling project and consult the official documentation for a more comprehensive understanding of each platform's capabilities and limitations.

ABM Platform	Features	Limitations	Model Scale	GIS Capability	Programming Language
NetLogo	- User-friendly interface and modelling environment -Large library of pre-built models - Graphical visualization - Supports agent interactions and complex behaviours	- Limited scalability for large-scale models - Limited support for advanced statistical analysis - Steeper learning curve for advanced customization	Small to Medium	Limited GIS capability, can import GIS data but lacks advanced GIS analysis	NetLogo
AnyLogic	- Multi-paradigm modelling (ABM, discrete event, system dynamics - Supports Java programming - Visual modelling with drag-and-drop components - Advanced visualization and animation capabilities	- High computational resource requirements - Requires knowledge of Java programming for advanced customization - Expensive commercial license for advanced features	Small to Large	Supports GIS data import and analysis through the GIS module	Java
Repast	- Flexible and customizable modelling framework - Supports Java programming - Modular design for easy integration of external libraries - High-performance computing capabilities	- Steeper learning curve for beginners - Requires programming skills for model development - Limited graphical interface compared to other platforms	Small to Large	Supports GIS data import and analysis through external libraries/plugins	Java (RepastS, RepastJ); Python (RepastPy); Visual Basic, .Net, C++, J#, C# (Repast.net)
GAMA	- Supports multiple modelling paradigms (ABM, cellular automata, system dynamics - Graphical interface with drag-and-drop components - Multi-level and multi-scale modelling capabilities - Open-source and extensible	- Limited user community and documentation - Less mature compared to other platforms - Less support for advanced statistical analysis	Small to Medium	Built-in GIS capabilities for spatial analysis and visualization	GAML (GAma Modeling Language) for simulations, Java for extensions
Mason	- Lightweight and efficient ABM framework - Supports Java programming	- Limited graphical interface for model development	Small to Large	Supports GIS data import and limited GIS analysis capabilities	Java

	- High-performance computing capabilities - Modular design for easy customization	- Requires programming skills for model development - Less user-friendly for beginners			
SWARM	- Designed specifically for ABM - Built-in support for spatial and network modelling - Supports Objective-C and Python programming - Scalable and efficient simulations	- Limited graphical interface for model development - Requires programming skills for model development - Less active development community compared to other platforms	Small to Large	Supports GIS data import and spatial analysis capabilities	Objective-C, Java
FAME	- Agent-based modelling framework - Supports Java and Python programming - Parallel and distributed computing capabilities - Modular design for easy customization	- Steeper learning curve for beginners - Requires programming skills for model development - Limited user community compared to other platforms	Small to Large	Supports GIS data import and limited GIS analysis capabilities	Java, Python
MESA	- Python-based ABM framework - Supports object-oriented programming - Easy model construction and experimentation - Extensive documentation and tutorials - Active development community	- Limited graphical interface for model development - Less mature compared to some other platforms - Less support for advanced statistical analysis compared to specialized platforms	Small to Large	Supports GIS data import and analysis through Python libraries	Python

Table 2 A comparison of various Agent-Based Modeling (ABM) platforms and software

Agent-Based Modeling (ABM) has a wide range of applications across various fields. In social sciences, ABM is used to study social phenomena such as opinion formation, crowd behaviour (Chunlin He et al., 2010), and the spread of infectious diseases (Cuevas, 2020). Agent-Based Modeling (ABM) provides a way to simulate and analyze individual behaviours and interactions within social networks, offering insights into collective dynamics and social patterns. For example, in the paper of (Cuevas, 2020), an agent-based model for evaluating the transmission risks of COVID-19 in various facilities is presented. The model incorporates spatial patterns and infection conditions that influence agent interactions and transmission dynamics. Each agent has an individual profile that defines their social characteristics and health conditions, shaping their behaviour during interactions. By simulating various scenarios, the model enables the exploration of different coexistence conditions and identifies effective measures to mitigate transmission risks. In economics, ABM is utilized to model complex economic systems, including markets, financial systems, and consumer behaviour. It enables researchers to explore the effects of different economic policies, trade scenarios, and market structures on macroeconomic outcomes. As mentioned in (Negahban & Yilmaz, 2014), unlike conventional tools, agent-based modelling and simulation (ABMS) offer several advantages for marketing research. Firstly, ABMS adopts a bottom-up approach by representing individual agents with heterogeneous attributes and decision-making processes, addressing population heterogeneity in marketing. Secondly, it allows for explicit modelling of the consumer social network, enabling the study of the impact of social influences on market dynamics. Finally, ABMS has the capability to explain complex non-linear marketing patterns by capturing emergent phenomena resulting from the micro-level behaviour of consumers and their interactions. These features make ABMS a valuable tool for investigating and understanding the dynamics of marketing systems. ABM is also valuable in ecology, where it helps study ecological systems, wildlife populations, ecosystem dynamics, and the spread of invasive species. For instance, the study (Marley et al., 2017) focuses on the interactions between urban areas and bears and the impact of bear dietary choices on both humans and bears. An agent-based model was utilized to examine the effects of educating humans about waste management and bear deterrence methods on the frequency of bear incursions into urban areas. A study investigates the long-term effects of releasing captive-born individuals with varied life histories into the wild for conservation purposes using forward-time, agent-based models (Willoughby & Christie, 2019).

Four species were examined: coho salmon, golden lion tamarin, western toad, and Whooping Crane. The study measured the impacts of supplementation by comparing population size and genetic diversity in supplemented populations to unaltered populations after 100 years. To summarize, by simulating individual organisms and their interactions with the environment, ABM provides insights into the emergence of ecological patterns, species coexistence, and the impacts of environmental changes. Additionally, ABM is also utilized in disaster management to simulate evacuation processes (Zhang et al., 2014) and assess the resilience of critical infrastructure (Thompson et al., 2019). In the field of organizational management, ABM allows the study of workforce dynamics, team collaboration, and the emergence of organizational behaviour. In transportation, ABM is applied to transportation planning and traffic management. It allows researchers to model individual travellers. The decision-making processes and interactions between commuters and the transportation network can be unrevealed, which enables the evaluation of different transportation policies, the optimization of traffic flows, and the assessment of infrastructure improvements. These applications are just a few examples of the diverse implementation of ABM in transportation. The application of ABM in transportation will be discussed mainly in the following sections. In summary, the ability to represent and simulate complex systems at the individual level makes the agent-based model a valuable tool for studying and understanding various phenomena in fields such as sociology, economics, ecology, and transportation.

2.1.3 INTEGRATION WITH GIS

As stated in (Crooks, 2018), the integration of geographic information systems (GIS) and agent-based modelling enables the incorporation of intelligent agents within a realistic environment. The integration of agent-based modelling and geographical data can be visualized through a

Geographic Information System. The world's complexity is represented by layers, including the physical and built environment. These layers establish a simulation world boundary where agents can operate. ABM (Agent-Based Modeling) and GIS (Geographic Information System) are distinct software tools that can be utilized to address different types of inquiries, and they share methodological elements that place them within the broader framework of geo-computation (Davies et al., 2019). The similarities between these two platforms are obvious. ABMs often employ a gridded world composed of attribute-carrying "patches," similar to raster data in GIS, which can also represent polygon-like data through rasterization. Agents in ABMs are attribute-carrying objects, typically zero-dimensional and akin to point-like GIS data. In an ABM, a time step corresponds to rule-based calculations in GIS that update feature attributes. From a GIS perspective, an agent-based model can be viewed as a layer capable of leveraging raster and vector datasets, transforming both itself and the underlying data (Davies et al., 2019). There is an architecture consisting of advanced ABM, GIS, and external modules proposed by (Guo et al., 2008); the agent-based modelling approach allows for the integration of parameter values, rule-based models, and interactions with the environment. GIS provides geospatial information, including topography, land cover, zoning, transportation, and social factors, to confine the agents' behaviour within the study area. External modules encompass user interface, simulator, visualization, and analysis tools. The user interface enables parameter settings through a graphical interface, facilitating sensitivity testing. The simulator sets up and executes external simulation models to carry out various tasks. Visualization and analysis tools support the examination and investigation of system outcomes. The architecture allows for the incorporation of additional auxiliary tools by modifying internal parameters or the ABM model and leveraging different spatial information (Guo et al., 2008). In general, most of the ABM software which is integrated with GIS uses a similar architecture.

2.1.4 LIMITATIONS

However, ABM also has certain limitations. Developing an ABM requires careful calibration of agent rules and parameters to ensure their realism and accuracy. Nevertheless, many of ABM involve human agents that bring soft factors into the model and make it difficult to quantify and calibrate (Bonabeau, 2002). Validating and verifying complex ABMs can be challenging, especially when dealing with large-scale systems. Additionally, ABMs can be computationally intensive, requiring substantial computational resources and time for simulations. For instance, (Balbi & Giupponi, 2009) examined ABMs in the field of climate change adaptation and discovered that due to the level of model abstraction, which hinders model testing, half of the assessed papers did not involve validation and verification.

In conclusion, agent-based modelling (ABM) offers a promising approach to understanding and simulating complex systems. By focusing on individual agents and their interactions, ABM provides a microscopic lens through which system-level patterns and emergent phenomena can be observed. While ABM has advantages in capturing heterogeneity and exploring "what-if" scenarios, it also has limitations in terms of calibration, validation, and computational requirements. Adhering to protocols such as ODD enhances transparency and reproducibility in ABM studies. Additionally, incorporating frameworks like the BDI architecture adds cognitive aspects to agent behaviour. With the availability of ABM platforms like NetLogo, Repast, and GAMA, researchers and practitioners have diverse tools to develop and analyze agent-based models. As ABM continues to evolve, it holds great potential for addressing complex challenges in various domains and guiding evidence-based decision-making.

2.2 AGENT-BASED MODEL IN TRANSPORTATION

The four-step travel model is a widely used tool for forecasting transportation demand and performance at a regional scale. It is primarily designed for evaluating large-scale infrastructure projects and comparing alternative interventions. However, it may not be suitable for capturing the complexities of managing existing infrastructure or implementing policies that directly influence travel behaviour (McNally, n.d.). The application of travel forecasting models is an ongoing process that requires continuous data collection, model estimation, and forecasting. However, limited time is available for systematically validating the accuracy of these models after their implementation. Additionally, as stated in (McNally, n.d.), the four-step model requires a lot of data to characterize the activity and transportation systems. It focuses more on the transportation planning side and provides a macroscopic view of the system. The four-step model relies on assumptions and aggregations that may not fully capture the complexity of individual travel behaviours. In contrast, the data needed for an agent-based model simulation focuses more on attributes and behaviour data at the individual agent level to simulate commuters in the real world. Agent-based models provide a platform for meeting the need of modern transportation simulation as they are capable of modelling real-world complexity through their modularity and flexibility, integrating different transportation-related models as required into a single framework. The incorporated transportation-related models might include models of land use changes, activity schedules, commercial location choice, housing location choice, mode choice, car ownership, road pricing, etc. (Kagho et al., 2020). In particular, with the introduction of new modes of transportation and technologies like autonomous vehicles, urban air mobility, route guidance technology, mobility as a service (MaaS) applications, and others, the focus has shifted from increasing transportation infrastructure to the present-day focus on travel demand management

(Kagho et al., 2020). Due to the autonomy of the agent in the simulation, even two agents from the same species might make different decisions when they face the same situation. This feature makes the agent-based model more suitable for modelling different travel demands with varying modes of transportation. The summary of the advantages of using agent-based modelling in transportation compared to conventional approaches are listed below (Bazzan & Klügl, 2014):

- It allows for the representation of heterogeneous and variable structures in both the agent population (such as individual drivers with different characteristics) and the transportation network (where links can be dynamically created or deleted). This flexibility enables a more realistic representation of the system.
- An agent-based approach enables the modelling of complex information processing and decision-making processes. Agents can consider multiple factors, anticipate future events, exhibit group behaviour, and adapt and learn from their experiences. This feature allows for a more comprehensive and dynamic representation of decision-making processes in transportation.
- An agent-based approach facilitates the integration of behavioural constraints throughout different levels and phases of the decision-making process. This means that the model can capture the influence of various factors on individual and collective behaviour, resulting in a more realistic simulation of traffic dynamics.

There are many choices of simulation platforms for ABM in transportation, and each of them has its advantages. As discussed by (Kagho et al., 2020), creating a multi-agent simulation for an agent-based model involves creating a transport network, introducing agents, and giving them rules on how to behave on the network based on real-life scenarios. In this work, several transportation simulation frameworks were described, which include TRANSIMS, MATSim, SimMobility, and

Polaris. TRANSIMS is a Los Alamos National Institute project and is used for disaggregating modelling of travel behaviour on large-scale transport networks. MATSim is a mesoscopic traffic flow simulator used for dynamic traffic assignment. SimMobility provides a multi-scale simulation platform that covers interactions of land use, transportation, and communication, modelling millions of agents. POLARIS provides a plug-and-play system for legacy software in its framework.

The application of agent-based modelling (ABM) in transportation at a city scale offers a versatile approach to analyzing and addressing various aspects of urban mobility. ABM can be utilized to understand and simulate travel demand, capturing the complex interactions between individuals, their activities, and the transportation system. There is a research focus on the travel demand dynamics during the hurricane evacuation process (Yin et al., 2014). This paper introduces an agent-based travel demand model system designed for simulating hurricane evacuation scenarios. The system incorporates various decision-making processes related to evacuation, such as determining whether to evacuate or stay, selecting accommodation type and destination, choosing transportation modes and vehicles, and deciding departure times. In order to model households' travel demand during the evacuation process, an agent-based approach was employed. It considers the travel demand is driven by the goal of seeking safety and engaging in various activities related to the evacuation (Yin et al., 2014). In other words, the decision-making process is involved in achieving these goals and participating in activities that form the various travel demands. These agents' behaviours were described using a set of interconnected econometric or statistical models, which allowed for a comprehensive representation of the decision-making processes and behaviours of households during the evacuation. Additionally, ABM enables the examination of traffic flow dynamics on the macro level, which includes congestion patterns, bottlenecks, and the effects of infrastructure changes. For example, an Agent-Based Model (ABM) is developed to simulate traffic patterns in San Francisco. The model uses a detailed road network and captures realistic variations in traffic conditions (Zhao et al., 2019). This paper presents an agent-based macroscopic traffic simulation model for San Francisco, aiming to create a city-scale infrastructure resiliency tool. The simulation balances abstraction and detail to enable efficient analysis. Traffic is simulated through interactions between individual vehicle agents, incorporating complex human behaviour. Agent mobility is simplified based on the volume-delay relationship for efficient real-time modelling and decision-making. The model is trip-based but can be adapted for activity-based simulations in the future, given the detailed network representation and fast simulation speed (Zhao et al., 2019). ABM also facilitates the study of mode choice behaviours and the impact of transportation policies on modal shift and travel behaviour. For example, in (Zou et al., 2016), an agent-based model that focuses on travellers' choices of mode and departure time to address traffic congestion was proposed. This model considers the decision-making process based on imperfect information and bounded rationality. Individuals accumulate travel experience by monitoring performance information of the road network and other relevant conditions such as traffic management policies and strategies. Through a Bayesian learning process, travellers gain spatial and temporal knowledge. Meanwhile, the model incorporates the theory of search gain and search cost with imperfect information to determine when a traveller will initiate or stop searching before travel mode and departure time are defined. After the decision making whether the agent sticks with the current mode or searches for an alternative mode, the travel mode and departure time can be determined. Furthermore, ABM can be employed to assess the effectiveness of intelligent transportation systems, such as traffic signal control algorithms, dynamic route guidance, and ride-sharing services. For instance, in the research (Han et al., 2015), a multi-agent traffic simulation system implemented based on the NetLogo platform was developed. The system represents urban traffic elements such as vehicles, road sections, and intersections as agent models. Each agent possesses essential capabilities of knowledge acquisition, autonomy, interaction, and communication. The road agent model incorporates traffic flow forecasting to influence the actions of vehicle agents and assist intersection agents in traffic signal control. Intersection agents serve as abstract models of signal controllers and monitor the traffic situation at intersections. The signal control function within each intersection agent model analyzes real-time and predicted traffic flow data obtained from interactions with related road agents. In conclusion, by incorporating spatial and temporal dimensions, ABM provides a holistic understanding of transportation systems, aiding in the development of efficient, sustainable, and resilient urban transportation strategies.

In Agent-Based Modeling (ABM) within the context of transportation, the concept of origin and destination (OD) plays a crucial role in simulating travel behaviours and understanding transportation patterns. Origin refers to the starting point of a traveller's journey, while the destination is the endpoint. The interaction between origins and destinations shapes the overall transportation network and influences travel choices. There are several examples to generate or define origins and destinations in the transportation ABM. In the work of (Lu et al., 2018), the office and residential buildings are used to represent the trip origin and destination since this work puts more focus on the peak period traffic. The study (Chen et al., 2016) utilized 1413 traffic analysis zones (TAZs) within the 5-county region, along with individual trip tables categorized by the origin and destination zones. Research from (Kloppel et al., 2019) initially assembled the information from an extensive travel survey conducted in 2008 to represent the travel demand in Munich accurately. The data obtained from the public travel survey provides a precise depiction of the travel patterns and behaviours exhibited by the residents of Munich. This data is instrumental in defining both the starting points and destinations of their journeys.

2.2.1 FLEET MANAGEMENT

Agent-based modelling (ABM) has emerged as a valuable approach for vehicle fleet management. ABM allows for the simulation and analysis of complex interactions and decision-making processes among individual vehicles within a fleet. In this context, each vehicle is represented as an autonomous agent with its own unique characteristics, including location, destination, and operational constraints. Research from (Martinez et al., 2015) introduces a novel concept of urban shared-taxi services aimed at utilizing traditional taxi capacity more efficiently. The system operates on a sharing basis, where passengers agree to share the vehicle with others who have compatible trips. An agent-based simulation model is proposed and tested, incorporating rules for matching requests with shared taxis based on space and time criteria. The simulation model focuses on replicating a typical working day in a city and includes a road network where taxis operate, and clients are generated based on trip generation indicators. The model incorporates a dispatcher system that centrally manages the assignment of taxis to clients using information such as the location of shared taxi vehicles, their occupancy rate, and client locations. The model also accounts for the possibility of hailing a taxi on the street or going directly to a taxi stand. The structure of the model allows for multiple taxi-owning companies with different fleet sizes, which can be connected to various phone dispatching companies or operate independently. The model emphasizes the interaction between clients and taxis, simulating their connection and service provision. While the model does not include a dynamic traffic model, it assumes a fixed traffic model due to the focus on the taxi market changes rather than their impact on traffic conditions. In the simulation model, the road network includes link attributes that represent travel time for different periods of the day, capturing the dynamic nature of traffic in the urban system. The model assumes that taxi drivers are experienced and have knowledge of the road network, enabling them to choose the fastest route to their destination. To compute the fastest route, Dijkstra's algorithm is employed. It is important to note that the model assumes the variation in the number of taxis in service does not impact the predefined travel speeds on the network links. The simulation experiment used Lisbon as the case study area, which demonstrates the potential of shared taxis in improving mobility management, with significant fare and travel time savings for passengers while minimizing the impact on taxi revenues. After two years, there is another research that uses Lisbon as the case study area to assess the impacts of deploying a shared self-driving urban mobility system (Martinez & Viegas, 2017). This research focuses on analyzing the potential effects of implementing a shared and self-driving fleet of vehicles in a mid-sized European city. The study investigates two distinct self-driving vehicle concepts: the Shared Taxi and the Taxi-Bus. The Shared Taxi concept resembles a traditional taxi service where passengers are willing to take small detours and share their rides with others. The Taxi-Bus concept involves minibuses that operate as a dynamic bus service, requiring customers to pre-book their rides in advance and walk short distances to designated stops. The dispatch system is responsible for collecting and processing real-time information necessary for creating and monitoring trips. When a user requests a ride, the Dispatcher determines the most suitable car or minibus to match with the user's request. In making this decision, the Dispatcher considers a time-minimization principle that aims to minimize travel time not only for the requesting user but also for the existing passengers already in the vehicle. (Heinrichs et al., 2017) presents an integrated approach to modelling car sharing as a new mode within a travel demand model. The approach utilizes disaggregated car fleets with car-specific attributes to represent the car-sharing service. The necessary parameters for mode choice are estimated from various surveys and integrated into an existing multinomial logit model. The proposed approach is then applied to simulate the travel demand of a synthetic population in Berlin, Germany. A synthetic population is generated for Berlin by combining statistical data and using the Iterative Proportional Fitting (IPF) approach. This approach estimates joint distributions of household and person attributes to match real-world socio-demographic characteristics closely. The population is spatially distributed based on population density, and each individual is assigned specific socio-demographic attributes. The resulting synthetic population is stored in a database for simulating different scenarios (Heinrichs et al., 2017).

2.2.2 AUTONOMOUS FLEET

Agent-based modelling (ABM) can be utilized in the context of autonomous vehicle fleets to simulate and study their behaviour and impacts. This approach allows for the examination of how different factors and variables, such as fleet size, dispatching algorithms, pricing schemes, and mobility patterns, influence the performance of the autonomous vehicle fleet. In the virtual world, the movement of fleet agents can involve various factors, including passenger demand, traffic conditions, road infrastructure, and fleet management strategies. In other words, each autonomous vehicle agent might have its own rules and objectives, such as picking up passengers, navigating the road network, and optimizing its route.

When examining the implementation of agent-based modelling (ABM) in the context of autonomous vehicle (AV) fleets, a comprehensive framework can be followed. Firstly, the model should accurately represent the dynamics of the AV fleet system, considering factors such as vehicle behaviour, passenger demand, and traffic conditions. The agents in the model, representing individual vehicles, passengers, and infrastructure components, should possess realistic attributes, behaviours, and decision-making capabilities. The model should incorporate various interactions, including vehicle-to-vehicle communication, passenger-vehicle interactions, and interactions with the surrounding environment. In the developed autonomous vehicle (AV) fleets agent-based model (Lu et al., 2018), each commuter is characterized by their home and workplace locations, representing residential and office buildings, respectively. The population density is determined by the spatial distribution of commuters' home locations at the start of the simulation. Commuters travel between their homes and workplaces on weekdays, typically starting their commute between 6:00 and 9:00 a.m. in the morning and returning home around 4:00-6:00 p.m. in the evening. The departure times from home and workplace follow a normal distribution. The 20,000 commuters in the model have the choice between using a personal car or an autonomous taxi (aTaxi) for their transportation needs. At the beginning of the simulation, idle aTaxis are randomly distributed throughout the city. During the simulation, aTaxis park directly at the last passenger's destination if they are not assigned to the next trip. They pick up commuters from their homes and transport them to their workplaces or vice versa. The maximum capacity of an aTaxi is set to four passengers. Only passengers with the same trip starting hour have the potential to share a vehicle. In the work of (Lu et al., 2018), the travel speed of the vehicles in the model varies based on the time of day and the number of vehicles on the road. This variation is used to simulate realistic traffic congestion during peak hours. The travel speed depends on the number of vehicles on the road and the road capacity. To optimize the route, the aTaxi aims to deliver all onboard passengers to their respective destinations using the shortest distance. The optimized route is determined based on the highest speed coefficient that allows all passengers to reach their destinations efficiently.

Additionally, the ABM framework should account for the spatial and temporal aspects of the AV fleet, simulating realistic movement patterns and trip schedules. To ensure the model's validity, the ABM should be calibrated and validated using real-world data to capture the characteristics of the

AV fleet system accurately. For example, in the research of (Kloppel et al., 2019), the simulation model utilizes the BeeZero booking data as input and is validated by comparing the simulation results with the original booking data. The validation process ensures the accuracy and reliability of the simulation. Out of the 394 bookings in the original dataset, 374 were successfully simulated. For the 374 successfully conducted bookings, the simulation model accurately calculated the price, including the application of the price packages based on the "best-price-guarantee" policy. In the work of (Lu et al., 2018), the behaviour model of the commuting simulation was calibrated and validated using real-world data, which enhances the credibility and trustworthiness of the agent-based model and its results. The validation process focused on three key components: commute speed, commute time, and commute trips by time of day. Data from an Ann Arbor commuting survey were utilized to validate the model. Meanwhile, data from the 2009 National Household Travel Survey (NHTS) were used to validate the commute trips by time of day. Meanwhile, the framework should also allow for the exploration of different scenarios, enabling the assessment of the fleet's performance under various conditions and the evaluation of potential interventions or policies. In the study of (Chen et al., 2016), several scenarios are considered to assess the sensitivity of fleet operation metrics in different conditions. The first scenario is a non-electric shared autonomous vehicle (SAV) with a 400-mile range and a 15-minute refuelling time, serving as a reference case for comparison. The second scenario focuses on shared autonomous electric vehicles (SAEVs) with an 80-mile range and a 4-hour recharge time, similar to current models of electric vehicles on the market. The third scenario introduces fast charging for SAEVs which reduces the recharge time to 30 minutes. However, the range is limited to 64 miles to protect battery capacity. The last two scenarios explore the use of long-range SAEVs with a 200-mile range (4-hour recharge time) and with a 160-mile range (30-minute fast charge time). These scenarios allow for a comprehensive analysis of the impact of vehicle range and charging infrastructure on fleet operations.

Finally, the outputs of the ABM simulation should be analyzed and interpreted to provide valuable insights into the behaviour and impacts of the AV fleet, aiding in the decision-making process for fleet management and policy development. For example, (Chen et al., 2016) presents an agent-based modelling (ABM) approach to examine the operations of a fleet of shared autonomous electric vehicles (SAEVs) in a medium-sized metropolitan area. As discussed by (Chen et al., 2016), based on the analysis of trip distance and time-of-day distributions, the results show that fleet size is influenced by battery recharge time and vehicle range. An 80-mile range SAEV can replace 3.7 privately owned vehicles, while a 200-mile range SAEV can replace 5.5 vehicles with Level II charging. With Level III fast-charging infrastructure, these ratios increase to 5.4 vehicles for the 80-mile range SAEV and 6.8 vehicles for the 200-mile range SAEV. In conclusion, by following the comprehensive framework mentioned above, ABM can serve as a powerful tool for understanding and optimizing autonomous vehicle fleet operations.

CHAPTER 3: MODEL CONSTRUCTION

3.1 PROPOSED MULTI-AGENT MODEL

The study utilizes agent-based modelling to simulate the operation of the autonomous vehicle fleet. It analyses the potential of using autonomous vehicles (AVs) as a transportation mode for people's daily commuting demands. While human drivers in taxi services are focused on maximizing their own profits, they might neglect the optimization of the entire fleet, which can result in poor utilization of resources, congestion, and inefficient routing. In contrast, autonomous cars can be operated as a fleet owned by one business, which can be managed and optimized depending on certain conditions, such as the availability of vehicles, the volume of demand, and the times of the day. Then, better coordination of operations and efficiency can be achieved. As mentioned by (Loeb et al., 2018), one of the benefits of fleet automation is the ability of vehicles in the system to respond to all ride requests immediately upon receiving them. This ensures prompt and efficient service for passengers. In this thesis, the demand-responsive operation strategy is the major research topic. The demand-responsive operations allow autonomous taxis to adjust their schedules and dispatch locations depending on the demand of people agents. The AV in the fleet can then be dispatched to areas experiencing high demand or placed in regions with low demand to ensure that the whole fleet is operated effectively and efficiently. Additionally, optimizing the fleet size and deployment based on demand change can cut the operation cost and reduce idling time for the individual vehicle, as well as minimize emissions and traffic load.

As discussed in the previous chapters, agent-based modelling (ABM) is a powerful modelling technique that enables the creation of more realistic representations of agents and the environment in a system. It is crucial to employ methods that can effectively capture and recreate the characteristics of complex systems to comprehend the intricacies of our world. Modelling is an invaluable tool for understanding complex systems, which provides a theoretical description of how a system or process operates (Crooks, 2018). A transportation system is complex and dynamic, with numerous participators with varying preferences and behaviours. With the help of ABM, the interactions between individuals in the system can be depicted in a way that is close to realistic and flexible, which makes it an ideal tool for transportation planning and analysis. The observers are able to build and investigate the various decision-making processes and behaviours of individuals and groups in a transportation system. The ability to create complex representations of the interaction between individual agents and the dynamic transportation system is enabled by defining the parameters and rules of behaviour for agents and the environment properly. After proper definition, the agents in the ABM are driven by rational behaviour by themselves to simulate real-world conditions and interactions. In other words, there are no complicated mathematical equations to define the behaviour of the agents and the parameters of the environment, which is the most different feature that ABM has compared to the traditional modelling technique.

For conducting the simulation-based case study in this thesis, the geospatial data for the target area needs to be acquired and form the base layers of the simulation. Geospatial data in this research refers to the information that can be collected and analyzed about a city's road network, neighbourhood area, and land use. A road network is composed of information about the connectivity, layout, and attributes of a city's roads and highways, such as intersections, road segments and hierarchy. It can also be used to improve the efficiency of a city's transportation management. The boundaries, characteristics, and spatial extent of each district or neighbourhood within a city are included in the neighbourhood data. This data may encompass demographic statistics and other relevant data that can provide insight into the community. The land use data includes information about the types, spatial distribution, and attributes of various land uses within a city, such as industrial, commercial, residential, and parks. By combining all the geospatial data and importing them into the ABM platform, the base layer of the simulation world can be assembled completely.

The framework of this simulation is formed by the city environment, people agents and fleet agents. Figure 3 illustrates a systematic framework for this thesis. As discussed in the previous paragraph, the city in the simulation consists of the road network, land use data and neighbourhood boundaries. The road network layer of the simulation incorporates the speed limit of each road segment, obtained from the city's data, as an attribute. The boundaries of the neighbourhoods define different statistical zones within the model. In this simulation, fleet agents are responsible for providing transportation services to the people agents who need to travel across the road network to reach destinations in various land-use areas for different purposes. When a person places a request, one of the fleet agents responds and fulfills the demand. The fleet agent picks up the person at the location of the request and drives them to their desired destination. In order to generate realistic demand patterns, data from the Transportation Tomorrow Survey (TTS) is utilized, capturing people's demand in different locations and during different time periods throughout the day. Additionally, the simulation dynamically adjusts the fleet size and deployment to reflect changes in demand.

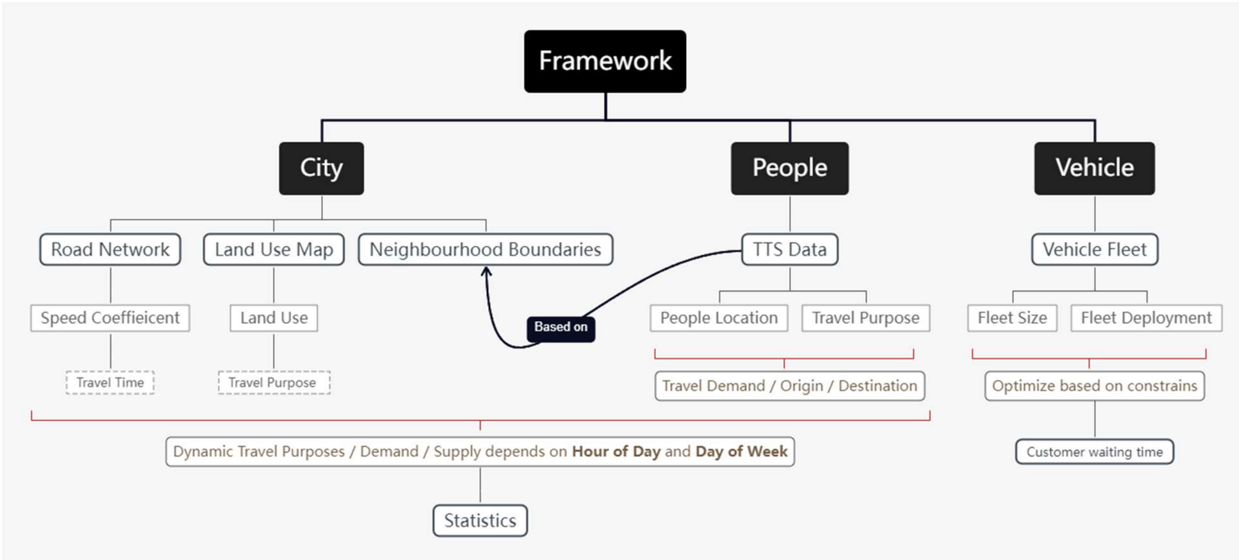

Figure 3 Simulation Model Framework

3.2 PLATFORMS

ABM platforms are software tools that provide researchers with the capability to develop, implement, and analyze agent-based models, such as model execution, data visualization and analysis. The integration of Geographic Information Systems (GIS) with Agent-Based Modeling (ABM) platforms allows spatial interactions and dynamics involved in the simulation and analysis of autonomous fleet operations. There are three major software tools that are involved in this thesis: Python GeoPandas, QGIS, and GAMA platform.

The open-source Python library GeoPandas is designed to help users work with spatial data, which includes information such as geographical coordinates and boundaries. It extends the Pandas library’s capabilities to handle spatial data, which makes it easy to perform various tasks related to spatial data, such as spatial join and network analysis. In this thesis, GeoPandas is involved in raw data processing and cleaning tasks to make the raw geospatial data ready for simulation use. It performs major spatial data editing tasks.

The QGIS (Quantum Geographic Information System) software (https://www.qgis.org/) is a widely used open-source GIS program that can be applied for spatial analysis, mapping, and visualization. Users can create, manage, and export geodata in different formats, such as CSV, shapefiles, and geodatabases. It also lets users manage spatial relationships and attribute data. QGIS has a wide range of tools for performing various spatial analysis tasks, which include geoprocessing, spatial queries, and buffers. These tools can help users extract insights from the data. After data cleaning procedures in this work, QGIS visualizes the spatial data and performs minor spatial data editing tasks.

The GAMA (Geographically and Agent-Based Modeling Architecture) platform is an open-source model and simulation framework that enables creating and analyzing agent-based models with spatial or geographic components. It supports the integration of various spatial and geographic data sources into its models, such as shapefiles, OSM files and Geotiff, which allows users to perform analysis and modelling of processes in a spatial environment (Taillandier et al., 2019). As stated by (Taillandier et al., 2019), the GAMA framework utilizes a dynamic quadtree structure that updates according to the agent's movement. It can also improve the spatial query and the shortest paths on graphs by implementing various algorithms such as the Dijkstra algorithm and the Floyd Warshall algorithm. The GAMA platform is the primary software that is employed in this work and carries out the experiment.

3.3 ODD PROTOCOL

3.3.1 OVERVIEW

Purpose: The purpose of this study is to investigate the improvement of fleet size and deployment in autonomous vehicle systems based on demand fluctuations. The research aims to explore how adjusting the fleet size according to real-time demand can lead to cost savings, reduced idle time, emission reduction, and improved traffic management. By analyzing the dynamics of demand on spatial and temporal scales, the study seeks to provide insights into the variation of the fleet size required to meet evolving transportation needs. The research aims to contribute to the development of more efficient, sustainable, and accessible autonomous vehicle systems. Furthermore, the ability to deactivate idle vehicles contributes directly to the reduction of emissions and the alleviation of traffic congestion. As inactive vehicles are taken out of circulation, the overall carbon footprint decreases, promoting a cleaner and greener environment. By strategically managing the fleet's deployment, traffic load can be efficiently distributed, preventing overcrowding on certain routes and minimizing delays for passengers.

Entities, state variables and scales: There are five types of species in this model: neighbourhoods, buildings, road networks, vehicles, and people. The entities and their related state variables are listed in the appendix. In this model, the people species has five objectives, which are “Home,” “Work,” “School,” “Discretionary,” and “Non-home based.” These five objectives indicate the purpose of travelling and the location of the people. The fleet species also has five objectives: "not in service," "searching," "pickup," "commuting," and "arrived." These objectives describe the different states of the vehicles in the simulation. The time step can be determined by users through the user interface. The default time step setting for this model is 1 minute, and the simulation time starts from 0 a.m. The virtual world in the model is constructed using imported shapefiles, which define the neighbourhoods, road networks, and buildings. The size of the virtual world depends on the size of the imported shapefiles, providing a realistic and customizable environment for the simulation.

Process overview and scheduling: According to TTS, people generate travelling demand at different times of the day. For example, there are 24% of travelling demand in a day happens in the morning rush hour, which means a specific number of people agents are generated during that period. The exact request hour and minute during this period for each people agent are randomly assigned once they are generated. Furthermore, the objectives of the people agent are assigned according to the different destinations. When the simulation time matches a people agent's request time, their boolean value “request” is set to true, indicating an active travel request. At this point, the fleet agents on the map detect the presence of active requests. The closest available fleet agent or the designated fleet agent is then assigned to the corresponding people agent. The matched fleet agent approaches the location where the active request is, and its objective changes to "pick up" at the same time. Once the fleet agent reaches the people agent and begins heading towards the people agent's destination, its objective changes to "commuting." Upon reaching the destination, the fleet agent's objective is updated to "arrived." The next objective for the fleet agent could be "searching" or "not in service," depending on the availability of new travel requests.

3.3.2 DESIGN CONCEPT

Emergence: The model aims to simulate the emergence of a dynamic autonomous vehicle (AV) system within a virtual environment. By simulating the interactions between people agents and fleet agents, emergent behaviour emerges as a result of their collective actions and decision-making processes. The model captures the dynamic nature of travel demand, fleet availability, and objective-driven interactions, leading to emergent patterns and outcomes.

Sensing: Both people agents and fleet agents have sensing capabilities within the simulation. People agents sense the current simulation time to determine if it matches their request time, triggering their travel demand. Furthermore, people agents are assigned a patience value once their travel request becomes active. The patience value represents their tolerance for waiting and influences the triggering of their decision to leave. People agents continuously sense and monitor the elapsed waiting time and patience value. If the waiting time or the patience value exceeds their threshold, they may choose to leave and seek alternative transportation options. The interplay between patience, waiting time, and the decision to leave adds a realistic and dynamic element to the model, capturing the varying levels of tolerance among individuals in the simulation. Fleet agents sense the presence of active travel requests and their proximity to these requests. Additionally, fleet agents can sense the status of their current objective, allowing them to transition to the next objective when certain conditions are met. Each fleet agent can sense the number of active members and compare it to the base population. As the fleet operates as a collective entity, fleet agents can communicate and share information with each other. By considering the constraints of searching time and idling time, as well as the base population setting, each fleet agent can make informed decisions and determine its objective. In addition, the model updates the information regarding potential high-demand areas based on different time periods. This information is then used to inform the fleet agents' decision-making process, enabling them to strategically allocate their resources and adjust their plans accordingly. As a result, the fleet agents can optimize their operations and adapt their behaviour based on the current demand and resource availability. The interaction and exchange of information among fleet agents contribute to the overall efficiency and effectiveness of the fleet management system.

Interaction: Interaction occurs between people agents and fleet agents in a coordinated manner. When an active travel request is detected, the closest available fleet agent or the designated fleet agent interacts with the corresponding people agent to provide transportation services. This interaction involves fleet agents approaching the pick-up location, picking up the people agent, and commuting toward the destination. The interaction between the two agent types is essential for fulfilling travel demands efficiently and effectively.

3.3.3 DETAILS

Initialization: The model follows an initialization phase to set up the virtual world and initialize the entities within it. This phase involves importing shapefiles to define the spatial elements of the simulation, such as neighbourhoods, road networks, and buildings. These shapefiles serve as the foundation for creating a realistic and geographically accurate environment. During this phase, the fleet agents and people agents are also initialized. They are assigned their initial attributes and states, such as objectives, destinations, and initial positions. This allows for the simulation to start with a predefined configuration of fleet agents and people agents, ready to interact and operate within the virtual world. The initialization phase ensures that the simulation begins with a well-defined and consistent state, providing a starting point for the subsequent simulation steps. The model establishes the foundation for the emergent behaviours and interactions that will unfold throughout the simulation by properly setting up the virtual world and initializing the entities within it.

Input: The model relies on input data to simulate the dynamics of the system. This input comprises shapefiles that define the spatial characteristics of the neighbourhood, road network, and buildings within the virtual world. These shapefiles serve as the foundation for creating an accurate and realistic environment for the simulation. Furthermore, the model incorporates TTS data, which provides essential information on travel patterns and demand at different times of the day. By utilizing this data, the model can generate travel demand in a proportional manner, accurately reflecting the real-world distribution of travel activities throughout the day. This ensures that the simulation captures the variability and patterns of travel demand based on temporal dynamics. By integrating both spatial and temporal data, the model creates a comprehensive representation of the transportation system, enabling the exploration and analysis of various scenarios and strategies. This input-driven approach enhances the accuracy and realism of the simulation, facilitating insightful observations and informed decision-making in transportation planning and management.

Submodels: The model consists of several interconnected submodels that simulate different aspects of the transportation system:

- People Generation: This submodel generates people agents based on the TTS and assigns them objectives and destinations according to their travel purposes (e.g., home, work, school). The request time for each person is randomly assigned within the corresponding demand period.
- Fleet Allocation & Distribution: This submodel manages the allocation of fleet agents to active requests and the distribution of fleet agents. When a request becomes active, fleet agents evaluate their proximity to the request location and their availability status. The closest available fleet agent or the designated fleet agent is assigned to the corresponding people agent, and their objectives are updated accordingly (e.g., from "searching" to "pick up"). The submodel also maintains the supply of fleet agents based on the average supply density in different neighbourhoods. By periodically updating the supply information, the submodel ensures that fleet agents are appropriately distributed to meet the demand in each area.
- Demand and Supply Management: This submodel tracks the demand and supply dynamics within the system. The fleet agents communicate and share information with each other to collectively manage the available supply. Based on constraints such as

searching time limitation and base supply population, the fleet agents make decisions regarding their objectives and resource allocation.

3.4 MODEL

Building an agent-based model (ABM) from the ground up involves several steps, which are listed below:

- Defining the problem: The definition of the problem involves identifying the model's objectives and system. It also entails identifying the environment, the agents and their behaviours.
- Implementing the model: The model should be implemented using a suitable programming language. As the previous section compared, a wide range of ABM platforms are available, such as GAMA, NetLogo, and AnyLogic. In this thesis, the GAMA platform was chosen.
- Testing the model: The model should be tested to ensure that it performs as expected. This process may involve parameter improvement, model validation, and sensitivity analysis.
- Experiments and analyzing the results: The analysis of the results should be carried out to interpret the model's findings and draw conclusions about its operation. This may involve scenario analysis, statistical analysis, or visualizations.

The following steps need to be followed to build an ABM on the GAMA platform specifically:

- A new project needs to be created.
- Defining the global environment for the model involves declaring the global environment parameters and defining the environment's behaviour and any external factors that may affect the agents.

- The species in the environment are described by specifying agent attributes, behaviours, and interactions with other agents or environments. Inside a species, the specie level parameters need to be declared at the beginning and then can be implemented in the agent behaviours and interactions.
- The initial conditions for the environment and species are defined by importing data or initial parameter settings. Before running the simulation, the user interface and display settings need to be settled.
- The GAMA's simulation engine runs the model and analyzes the results by using built-in analysis tools or exporting data to an external program for further analysis.

3.4.1 GLOBAL

As the steps listed in the previous section, the global environment needs to be defined at the beginning of the simulation. After data preparation, the shapefiles are ready to import into the model, which forms the buildings, roads, neighbourhoods, and boundaries in the simulation environment. The shapefile of city roads also defines the graph for the road network in the simulation, which declares the data type (graph) of the road network in the global environment. Time and date variables are defined, including steps for the model (time interval between two simulation cycles), starting date, and boolean variables for different time periods (AM peak, mid-day, PM peak, evening, and early morning). The AM peak starts from 7 to 10, and the PM peak begins from 16 to 19. Mid-day is defined as between 10 to 16. After the PM peak till 0 a.m. is the evening, and from 0 a.m. to the start of AM peak is the early morning. The day and night boolean variables are also declared. Meanwhile, all of the global variables are declared before initiation, such as agent population and some of the agent behaviour variables.

The 'init' statement in the GAML language serves as the initialization step for the simulation. It is responsible for defining the initial values of environment elements and agents that are necessary for the simulation process. In this simulation, the 'init' statement is used in the global environment to create the initial species of the model. Specifically, it generates the species for neighbourhoods, buildings, and the road network. Additionally, the statement randomly deploys the initial fleet agents on the road network. In order to ensure that the fleet agents choose the shortest path when transporting the people agents to their destinations, the A-star algorithm (Hart et al., 1968) is defined and applied to the road network in the model during the initial stage. Each road segment is assigned a weight based on its perimeter, allowing for efficient path selection by the fleet agents.

In the GAML, behaviours, also known as reflexes, are sets of statements that agents execute at each step of the procedure. The "when" method, a facet of the behaviour, allows the reflex to be executed only when a specific boolean expression evaluates to true. This capability simplifies the specification of agent actions and decision-making processes (Taillandier et al., 2019). In this particular simulation, the vehicle agents represent the supply side, while the people agents represent the demand side. The simulation records the number of vehicles in service and the number of waiting people. Additionally, it calculates metrics such as the average wait time for people and the number of waiting people within a specific period. To determine the speed of vehicles, the "speed_coeff" and "speed" reflexes are utilized. These reflexes adjust the speed of vehicles based on the type of road and the time of day. The speed coefficient is assigned a different random value within a specified range depending on the time of day, allowing for realistic variations in vehicle speeds.

The supply side of the simulation involves defining the base populations of vehicles for different times of the day. The "base_supply_needed" reflex determines whether the current number of vehicles is sufficient to meet the base population requirements of the in-service fleet. Figure 4 presents a flow chart illustrating the workflow of the reflexes responsible for maintaining the supply number and dispatching supply to different areas. The average number of vehicles per square kilometer in each neighbourhood is calculated. Three list variables are declared in the global environment to facilitate this process. These variables are used to monitor and collect information about neighbourhoods with zero supply, as well as those with supply above or below the average number. The "maintain_supply" and "dispatch_supply" reflexes play a crucial role in ensuring that the number of vehicles in service matches the base population requirements. These reflexes work together to dispatch standby vehicles from neighbourhoods with an above-average supply to those experiencing a shortage of supply. This mechanism helps balance the distribution of vehicles across different areas. On the demand side, the behaviours appear to define the generation of weekday demand for different periods of the day based on the type of travelling (home base and non-home-based) and percentage of people interested in different activities (e.g., work, school, and discretionary). The data is extracted from the Transportation Tomorrow Survey (TTS). There is an example of data that is represented in Figure 5. In this table, the rush hour data is detailed, and the percentage of the rush hour trips in total trips in 24 hours is indicated. The trip purpose has two major types, which are home-based trips and non-home-based trips. Based on the destination of the home base trip, they are classified into three types: home base to work, home base to school and home base to discretionary places.

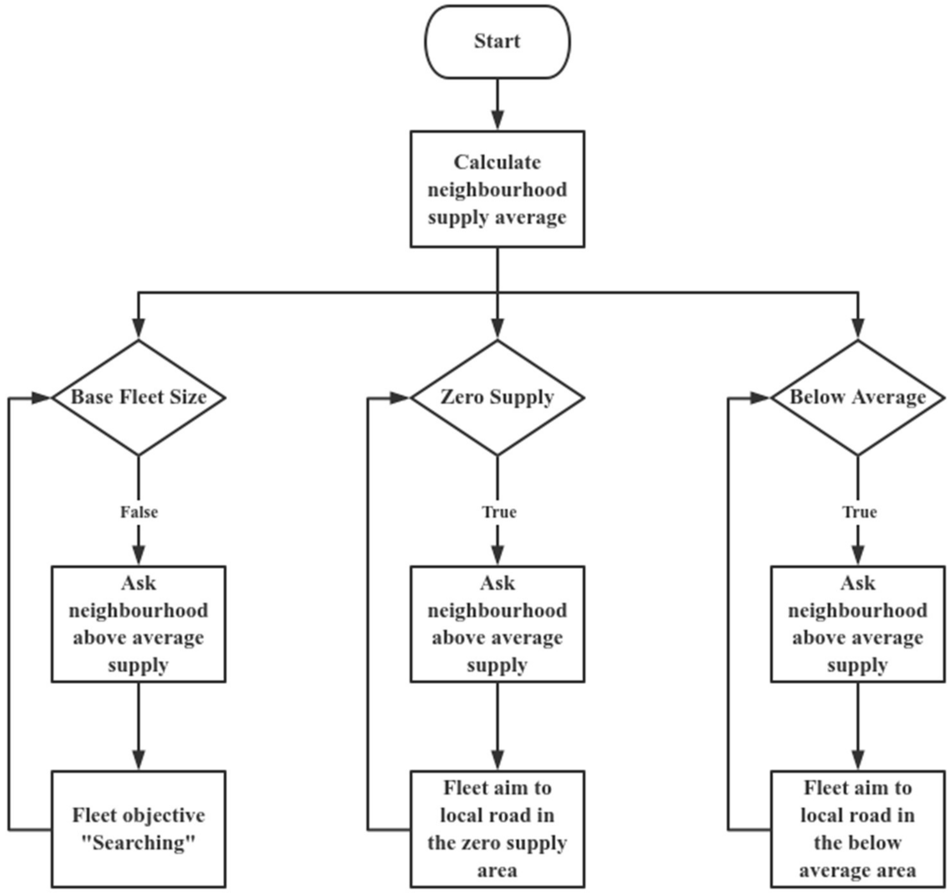

Figure 4 The three major behaviours to maintain the base supply number and dispatching.

TRIPS MADE BY RESIDENTS OF FORMER CITY OF TORONTO																	
Time Period	Trips	% 24hr	Trip Purpose				Mode of Travel						Median Trip Length (km)				
			HB-W	HB-S	HB-D	N-HB	Driver	Pass.	Transit	GO Train	Walk & Cycle	Other	Driver	Pass.	Transit	GO Train
6:00-8:59 AM	371,400	23.1%	63%	13%	17%	7%	31%	5%	35%	0%	26%	2%	6.7	3.4	5.1	14.1
	326,700	22.4%	58%	18%	16%	8%	37%	8%	35%	0%	19%	2%	5.8	3.4	5.1	28.2
	299,600	22.5%	58%	19%	15%	7%	38%	8%	35%	0%	18%	1%	6.1	3.5	5.0	28.5
	283,300	22.3%	63%	20%	11%	5%	38%	9%	35%	0%	17%	1%	6.9	4.3	5.0	24.8
	265,500	24.3%	71%	16%	9%	4%	37%	7%	42%	0%	13%	1%	7.0	5.4	4.9	16.0
24 Hrs	1,610,100		39%	9%	36%	16%	32%	7%	31%	0%	27%	3%	5.2	3.9	4.6	19.1
	1,457,100		36%	11%	37%	16%	40%	10%	30%	0%	17%	2%	4.4	3.3	4.2	28.7
	1,332,300		37%	11%	37%	15%	42%	11%	30%	0%	15%	2%	4.4	3.6	4.5	28.1
	1,271,900		39%	12%	33%	16%	43%	12%	29%	0%	14%	2%	5.0	3.8	4.5	25.9
	1,095,000		45%	12%	30%	14%	42%	10%	36%	0%	10%	2%	5.1	4.2	4.5	25.0

Figure 5 The total trips, the percentage of trips in a day, and their purposes.

In this thesis, the simulation focuses on the weekday travel demand. Due to the lack of precise data for other times of the day, the trip data for the rush hour in the afternoon is duplicated from the rush hour in the morning. Same as the percentage of the total trip in the morning rush hour, 24% of the trip in a day is assigned to the afternoon rush hour. The rest of the trip is distributed to the other three times of the day rationally: 24% of trips happen in the mid-day, 20% in the evening, and 8% in the early morning. Since the total trip and each percentage of trip purposes in 24 hours are known, the different trip purposes are distributed to the mid-day, early morning and evening according to their percentage of the total trip. In order to create some randomness for the demand generating, each demand in a certain period of the day is assigned a random request time in the period. It means the vehicle agents can only detect the demand when the simulation time matches the request time of the demand exactly. As Figure 6 shows, each time of the day has assigned a certain number of people according to the calculation based on 2016 TTS data. The origin and destination of people are defined based on 2016 TTS data as well. As discussed above, every single people agent at a time of the day has a random request time. When the simulation time matches the request time of the people agent, the request turns true request and can be detected by the vehicle agents.

The last behaviour of the global environment is updating the high-demand or potential high-demand neighbourhoods. A reflex is created and intended to identify high-demand areas ("hot areas") in a model. There are three main conditions that determine which buildings or neighbourhoods are considered "hot areas":

- If it is during the AM peak hour, the "hot buildings" are defined as all residential buildings.

- If it is during the PM peak hour, the "hot buildings" are defined as all commercial, employment, or mixed-use buildings (they are all generally workplaces).

If it is during any other time of day (is_early_morning, is_mid_day, or is_evening), the reflex considers each neighbourhood in the model. If a neighbourhood has active demands for more people greater than or equal to 10% of the total number of people in the period of the day, then that neighbourhood is considered "hot." If a neighbourhood has the potential to have more people greater than or equal to 1% of the total number of people in the period of the day, then that neighbourhood is also considered "hot."

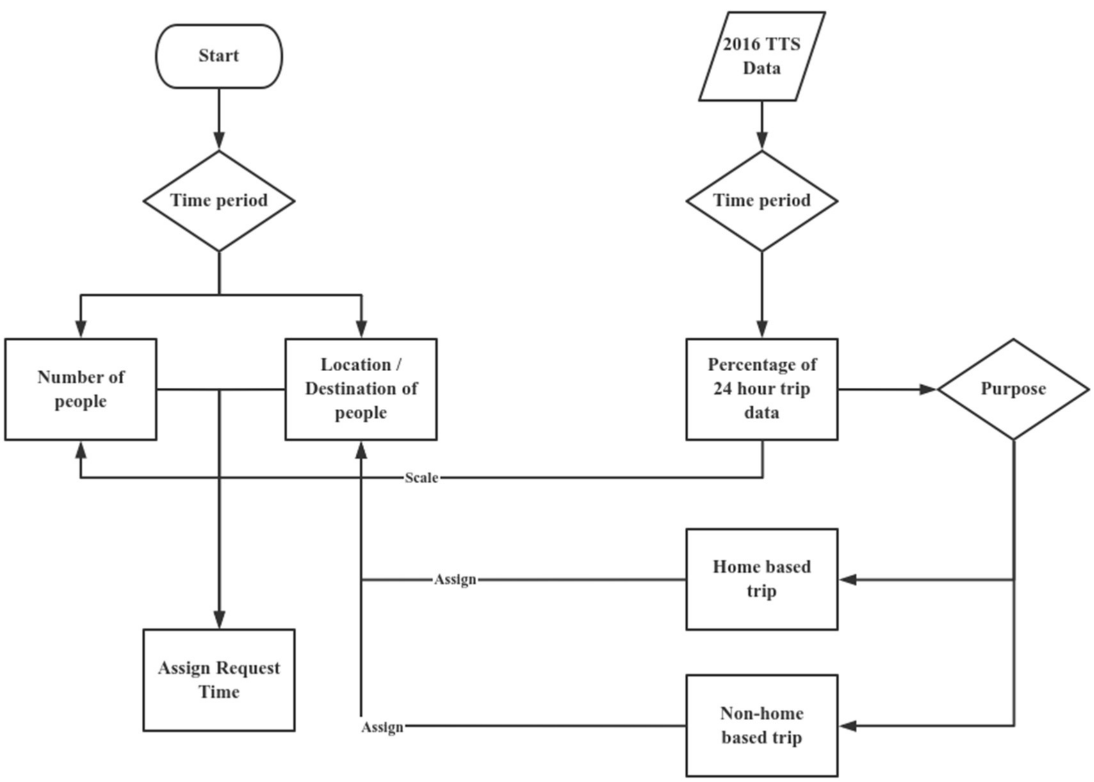

Figure 6 The generation process of the people agent (demand).

3.4.2 SPECIES

In GAML (Generative Agent-based Modeling Language), a species defines the attributes and behaviours of a group of agents. It serves as a blueprint for creating multiple instances of agents with similar properties and capabilities. A species can have various features in GAML. The main types of species' features are attributes and behaviours. The former defines the agent's characteristics, and these can be expressed using data such as integers, strings, and lists. On the other hand, the latter defines the actions of the entity. For instance, base elements can represent the creature's physical appearance, while a reflex aspect can specify its responsiveness to certain situations. The "neighbourhood" species is defined by the keyword 'species' declaration that represents a geographical area. The species contains several attributes, such as the area name, area square kilometers, and neighbourhood demand number. There are two reflexes defined in the species: "update_demand_number" and "update_supply_number." The former reflex updates the neighbourhood demand number and potential number by counting the number of people inside the neighbourhood with true and false requests, respectively. The latter reflex updates the neighbourhood supply number and supply square kilometers number by counting the number of fleet objects inside the neighbourhood with the objective "not in service." The building species has four attributes:

- type: a string that represents the building type (e.g., residential, commercial, employment, mixed-use, etc.).
- area_name: a string that represents the name of the neighbourhood where the building is located.
- move_id: an integer that represents the movement ID of the building.
- colour: an RGB colour that represents the colour of the building types in a visualization.

The building species is helpful in modeling the built environment in the simulation, as they play an essential role in determining the origin and destination of the travelling. Similar to building species, the definition of the road species is simple, and it has only one attribute called "type." According to different types of road segments, the speed limit is signed for each of them.

```
species neighbourhood {
    // Feature from GIS file - Area name / Movement id / Square meters
    string area_name;
    string uber_name;
    int move_id;
    int area_sqm;
    float area_sqkm <- area_sqm / 1000000;

    int neighbourhood_demand_num;
    int neighbourhood_potential_num;

    int neighbourhood_supply_num;
    int neighbourhood_supply_sqkm_num;

    reflex update_demand_number {
        neighbourhood_demand_num <- people inside self count (each.request = true);
        neighbourhood_potential_num <- people inside self count (each.request = false);
    }

    reflex update_supply_number {
        neighbourhood_supply_num <- fleet inside self count (each.objective = "not in service");
        neighbourhood_supply_sqkm_num <- neighbourhood_supply_num / area_sqkm;
    }

    aspect base {
        draw shape border: #orange width: 2;
    }
}
```

Figure 7 Example of the code of defining a species in GAML.

The people species defines a type of agent that can make requests for transportation services and ask fleet species to come and pick up. The people have various properties such as request time, current fleet, objective, and patience. The code includes reflexes that trigger certain actions based on specific conditions. For example, when a request is made, the reflexes update the area and fleet, calculate wait times, and set triggers for leaving the system if there is no match, or the patience value threshold is exceeded. Additionally, in the aspect statement, different colours are used to indicate whether a request is active or not. There are five different objectives that describe the state of each people agent: "Home," "Work," "School," "Discretionary," and "Non-Home-Based." These five objectives represent the purpose of the trip. Figure 8 is the flowchart that represents the workflow of people species. When the system time matches the request time of a people agent, this people's request turns active. After a certain time period, the patience value is decreasing in every minute, and the people agent has limited waiting time for different times of the day. In the AM or PM rush hour, the waiting time limit is shorter than at other times. Either the patience value is not enough or the waiting time limit is exceeded, the people agent will check if a fleet agent matches it and whether the distance between itself and the fleet agent is within 1 kilometer. If there is no fleet agent matches with the people agent, or there is a fleet agent matches with the people agent, but the distance between these two agents is more than 1 kilometer, the people agent will leave the system. As a result, this active demand will count as unfilled demand and will be displayed in the chart. On the other hand, if the fleet agent is close enough to the people agent, then the people agent will keep waiting for pick-up even if the patience value and waiting time limit have exceeded the threshold. The code that defines fleet species includes a set of skills that the species of fleet possess, such as searching (which is implemented using a series of actions and reflexes), navigating, and commuting. The fleet agents are able to pick up and transport people (defined by the "people" species) and have a set of rules that govern their behaviour in certain situations, such as maintaining a larger fleet size during peak travel times or dispatch to specific hotspots (defined by the "hot_building" and "hot_neighbourhood" variables). There are five different objectives that describe the state of each fleet agent: "not in service," "searching," "pickup," "commuting," and "arrived." A typical workflow that was tested in the simulation is illustrated in Figure 9. The fleet agent keeps searching for customers in the high-demand area if it is not matched with a customer. Every time the fleet agent matches with a people agent, the search time will be renewed. However, if the searching time and the number of fleets in service are larger than the optimization time and the base population of the fleet in service, while the objective of the fleet agent is “searching,” the optimization behaviour will be conducted. Fleet agents who meet the criteria will park along the local road in a high-demand area and turn the objective to “not in service,” which means it is optimized out of the system and wait for active again by designated demand.

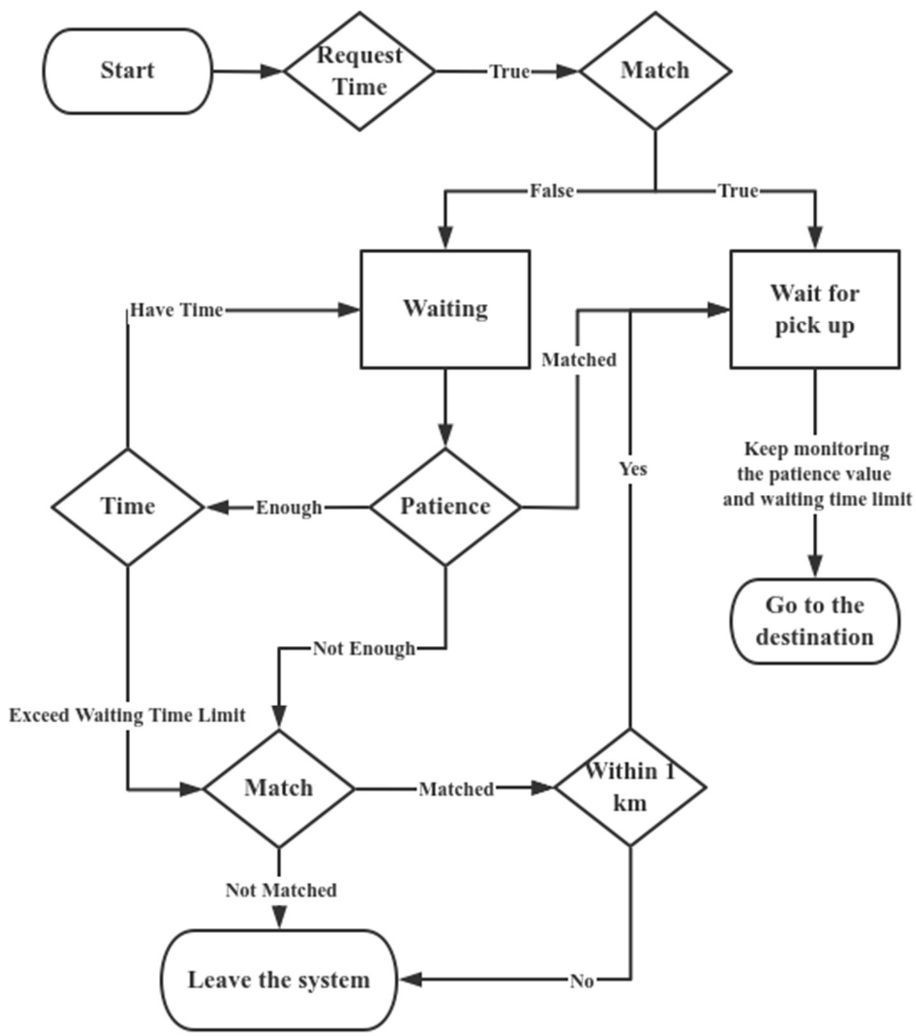

Figure 8 Flowchart of the working process of people species

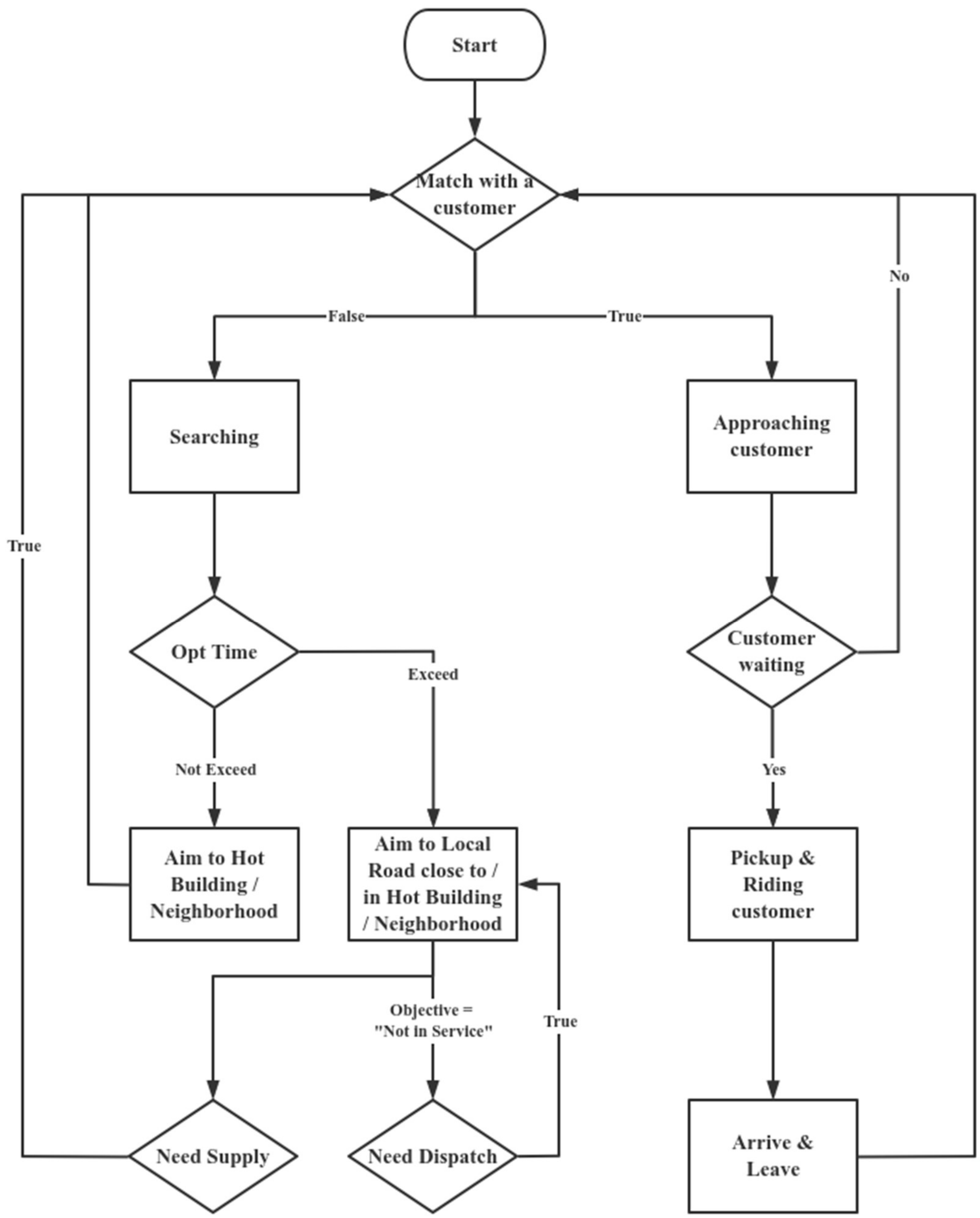

Figure 9 Flowchart of the working process of fleet species

3.4.3 EXPERIMENT INTERFACE

The GAMA platform integrated development environment (IDE) provides a graphical user interface (GUI) that allows users to monitor the simulation while it's running. The functions include adjusting model parameters, visualizing simulation results, and interacting with the simulation during runtime, which shows in Figure 10.

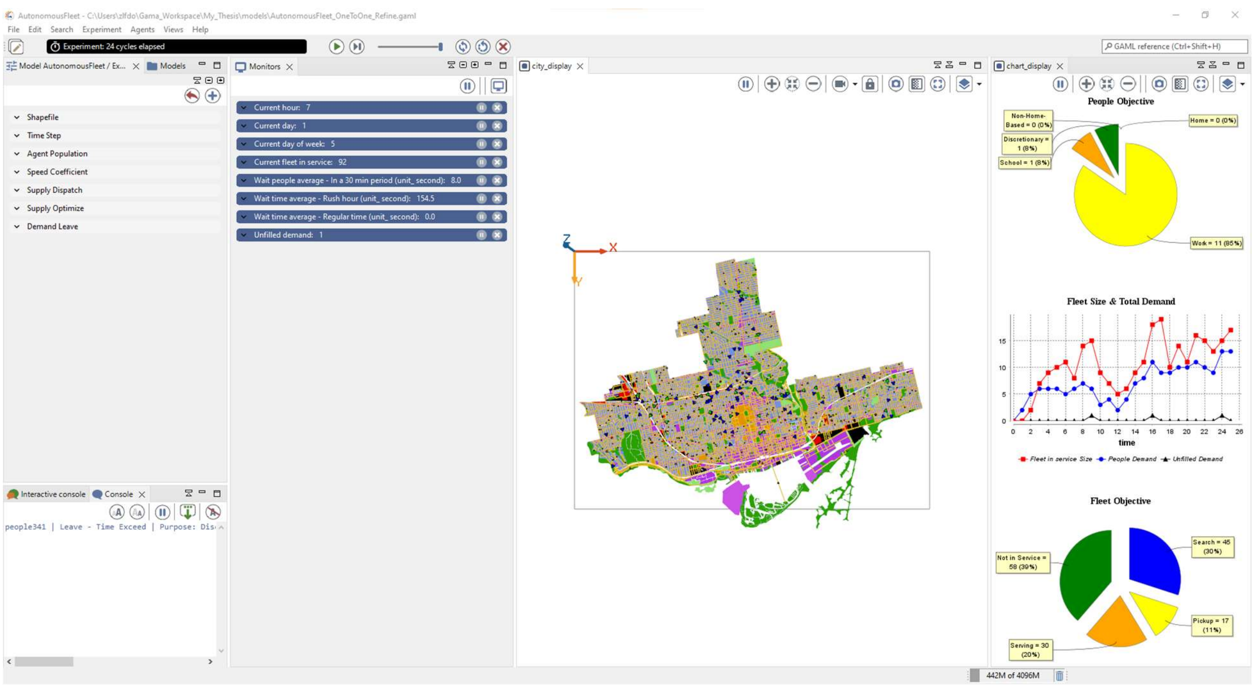

Figure 10 GAMA Platform graphical user interface (GUI)

The GAMA platform's GUI contains various panels that show different sections of the simulation and model, such as the input section and output sections. Firstly, it provides a comprehensive model console, which is the left part of the interface in Figure 10. Users can easily adjust parameters in the ABM world through the user inputs panel, such as time steps, agent initial population and speed coefficient, and see the updates in the console. **Error! Reference source not found.** is the screenshot of the model console.

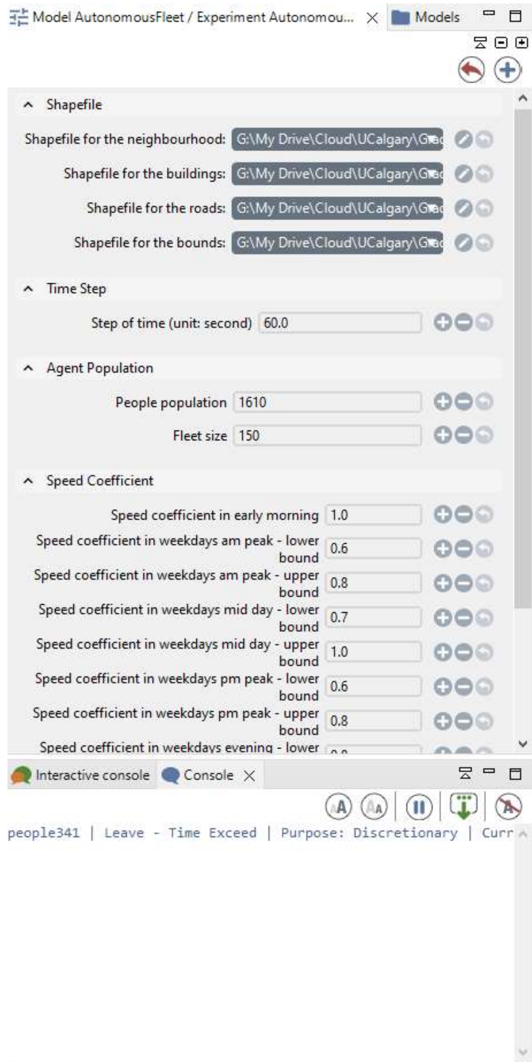

Figure 11 Model console

To monitor the simulation in real-time, the GAMA platform offers model monitors, which is the second part of this interface. The detailed screenshot is referred to in Figure 12. This allows us to observe the dynamic of agents and the environment. The monitor panel tracks various parameters related to the simulation, such as the current time, the number of vehicles in service, and the average wait times for customers in this simulation.

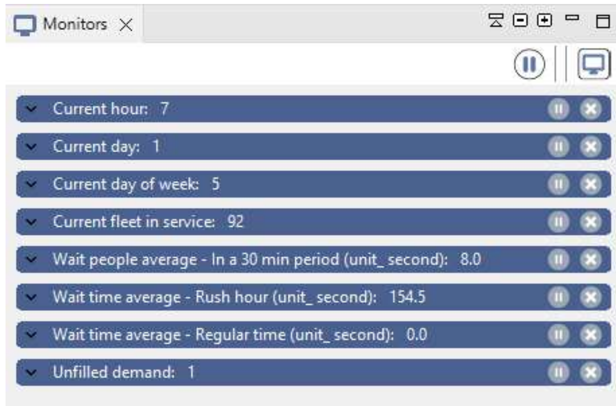

Figure 12 Model monitors

The third part of this GUI is a powerful model display feature. The visual representation enhances the understanding of the model and facilitates a more intuitive exploration of the simulated system. The display panel represents the geographical location of the agents and the simulation environment. The charts in the chart display panel show the distribution of people's and fleet's objectives and the fleet size and demand over time. Figure 13 and Figure 14 represent one of the charts from the chart display panel and model visualization.

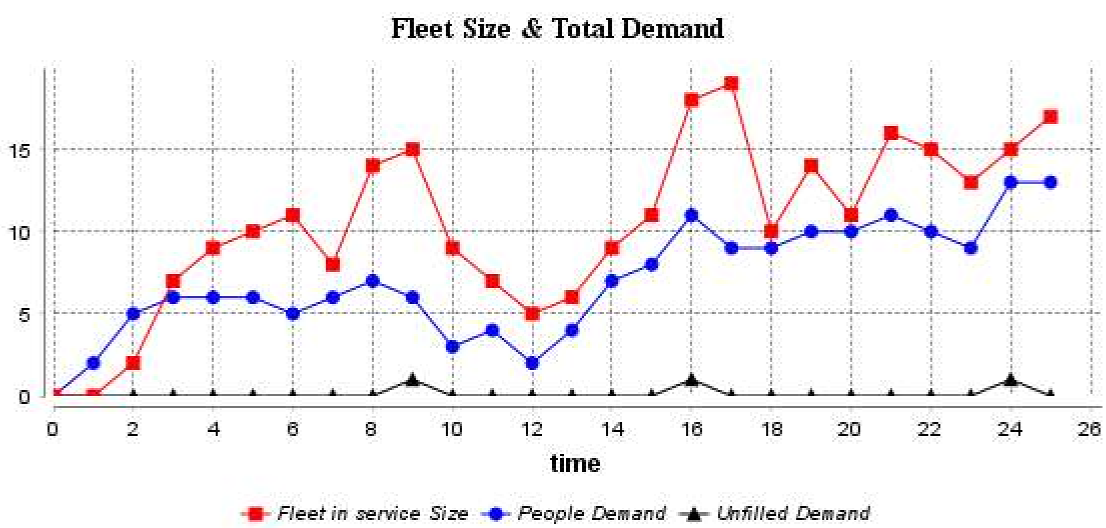

Figure 13 Fleet size and demand chart

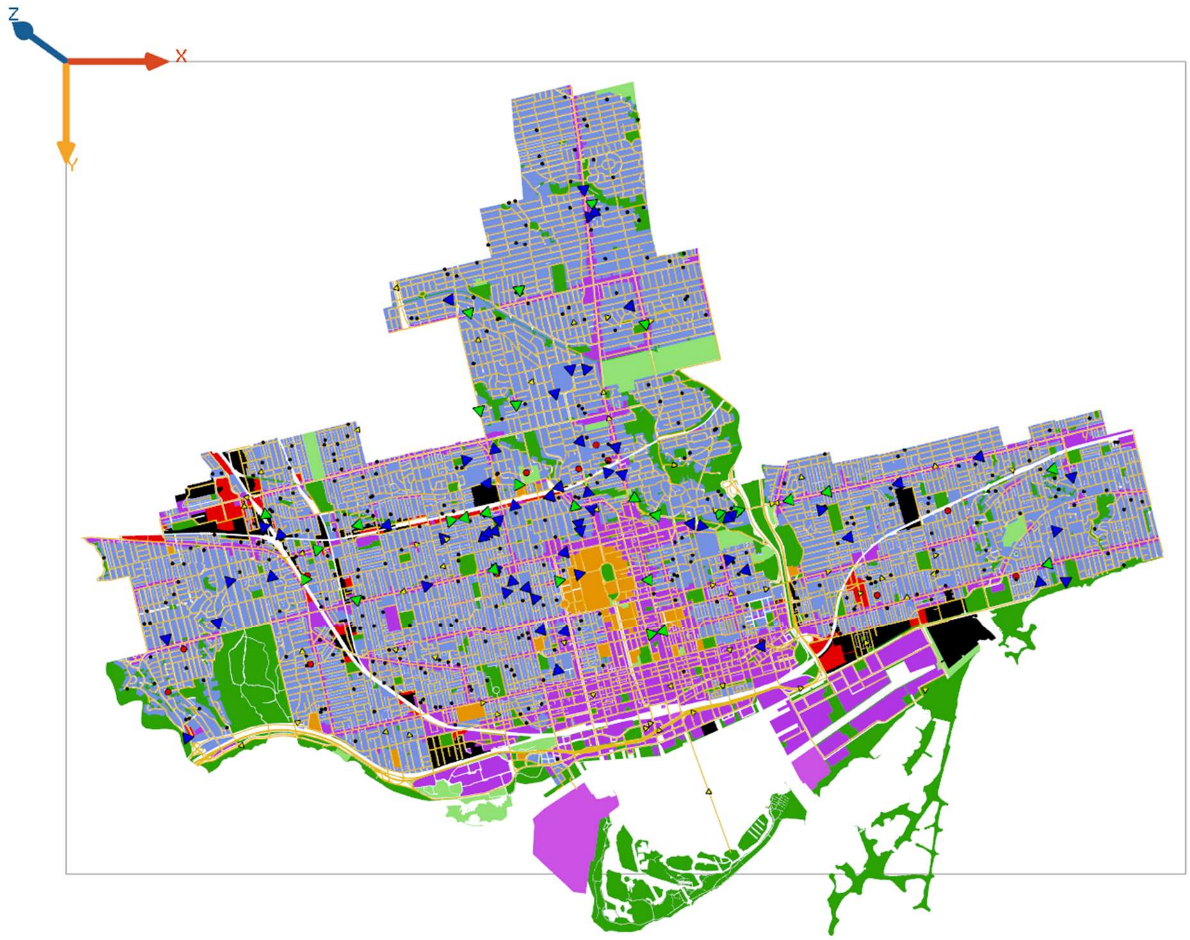

Figure 14 Model visualization and charts

CHAPTER 4: CASE STUDY AND EXPERIMENT

4.1 DATA

The City of Toronto, located in Ontario, Canada, is the study case in this thesis. Through its Open Data program, Toronto allows the public to access various datasets about its services and infrastructure. The neighbourhood boundaries and road network data can be acquired from the City of Toronto Open Data Portal. However, due to the lack of land use data (shapefile) in the Open Data Portal, this thesis uses the dataset named “Toronto Land Use Spatial Data - parcel-level - (2019-2021)” (Fortin, 2022) in the Map & Data Library Dataverse from the University of Toronto.

The raw data for neighbourhood boundaries and road networks, provided by the City of Toronto, contains various categories and attributes to cater to different usage requirements. Before importing the data into the GAMA platform for simulation purposes, the data attribute table undergoes a cleaning and editing process. This step focuses on addressing incompleteness and correcting any mistakes present in the dataset. As described in the research by (Fortin, 2022), the resulting layer, named "LanduseParcelsMergedv0120220106," undergoes a dissolution process based on the "Class_name" attribute. This operation combines land use areas with the same land use class name into multi-polygons representing the same category. This dissolved dataset serves as the raw data for land use in this thesis. To ensure the imported data in the GAMA platform is relevant and devoid of uncorrelated or duplicate attributes, additional data cleaning and fusion steps are performed. Specifically, the speed limit for each road segment in the road network dataset is assigned, and the original attribute table retains only the hierarchical attributes. Similarly, the neighbourhood boundaries data is modified to keep only one attribute column named "area_name" and includes a new column indicating the area of each neighbourhood in square kilometers. The land use data is reduced to two attribute columns: land use type and the name of the neighbourhood to which the building belongs. Overall, the data cleaning and editing processes are crucial to prepare the raw data for accurate and efficient utilization within the GAMA platform for simulation purposes.

4.1.1 DATA ACQUISITION AND CLEANING

At the stage of data acquisition, the QGIS is used to visualize the data and check the integrity and quality of the data. The OSM (Open Street Map) for the Province of Ontario was considered. After clipping to the research area and dropping the unnecessary attribute column, there is a lot of ‘Null’ value for the type of building. As Figure 15 illustrates that only small parts of the buildings in the city have the building type attribute data. Additionally, the zoning map that is published by the City of Toronto was considered as well. However, the data integrity cannot meet the requirements of this thesis since there is a large area of zoning spaces without any data that exist in the dataset. As a result, the building data is based on the dataset discussed in previous sections, which is “Toronto Land Use Spatial Data - parcel-level - (2019-2021)”.

The dataset of “Toronto Parcel-Level Land Use Spatial Data” has 495,875 rows of data in the attribute table, and all parcels that have the same land use type are aggregated. The spatial analysis function in QGIS named “Multipart to Single part” was applied to the aggregated dataset, and the total rows of data were reduced to 21,873 rows. Meanwhile, there are some issues that exist in the dataset. Firstly, the connections between some land use areas and others have not been corrected. Secondly, the classification of some land use areas has been misclassified, which can lead to inaccurate decision-making and data analysis. Several solutions were implemented to address these issues. One solution involves manually splitting features and editing attributes to ensure that land use areas are connected correctly and classified accurately. Another solution involves deleting extra parts that are not relevant to the analysis. Additionally, based on the dataset named 'Property Boundaries' that was downloaded from the City of Toronto open data portal, it is possible to create a new polygon for the misclassified area to ensure that it is properly classified and analyzed. Finally, the tool name "Check Validity" in QGIS was employed to guarantee the integrity of the dataset. It is a geoprocessing tool in QGIS that allows users to check the validity of a layer's geometry, which highlight errors that can affect spatial analysis, such as self-intersections and invalid shapes. After the processing, the geometry of the building layer was ready to import into the GAMA platform.

Figure 15 OSM building data with building type is not 'Null'

The neighbourhood data was acquired from the City of Toronto open data portal. These files contain information about the neighbourhood's unique identifier, name, and other attributes. During the cleaning process, some unnecessary attribute columns were dropped. The road centreline data of the City of Toronto is the primary resource for generating road network data. It provides the details about the name, hierarchy of each road segment and other attributes. The unnecessary attribute columns were also dropped during the cleaning process.

4.1.2 DATA MERGING AND EDITING

Spatial join is a common technique used in GIS to combine data from two different layers based on their spatial relationships. In this case, the goal is to join the land use dataset with the neighbourhoods dataset to assign each land use polygon to its corresponding neighbourhood. However, there are some challenges with performing a general spatial join in QGIS or GeoPandas. For example, some land use areas may fall within two neighbourhoods, and performing an overlap spatial join may not be possible based on the largest overlap area. One approach to address this challenge is using GeoPandas to get each polygon’s centroid in the land use dataset. These centroids can then be spatially joined with the neighbourhoods dataset while keeping the original geometry column for the land use dataset. After the spatial join is completed, the centroids can be dropped, and the geometry column can be recovered in the land use dataset. This approach allows for a more accurate assignment of land use areas to neighbourhoods, even when overlaps or multiple neighbourhoods are involved.

Road network dataset editing was done based on the City of Toronto Road Classification System (City of Toronto, n.d.), which provides information on the hierarchy of roads and their maximum speeds. Matching the maximum speed with the hierarchy of the road can provide real-world settings to the simulation environment and increase the accuracy of the simulation. As Figure 16 illustrates, the maximum speed assignment was realized by using Python code in Jupyter Notebooks in the Anaconda environment.

```
In [7]: gdf["Max_speed"]=""
        gdf=gdf[['FEATURE_00','Max_speed', 'geometry']]
        gdf.FEATURE_00.unique()

Out[7]: array(['Expressway Ramp', 'Collector', 'Local', 'Expressway',
               'Major Arterial', 'Minor Arterial', 'Major Arterial Ramp',
               'Collector Ramp', 'Minor Arterial Ramp'], dtype=object)

In [9]: gdf.loc[gdf['FEATURE_00'] == 'Expressway', 'Max_speed'] = 90
        gdf.loc[gdf['FEATURE_00'] == 'Expressway Ramp', 'Max_speed'] = 90
        gdf.loc[gdf['FEATURE_00'] == 'Major Arterial', 'Max_speed'] = 55
        gdf.loc[gdf['FEATURE_00'] == 'Major Arterial Ramp', 'Max_speed'] = 55
        gdf.loc[gdf['FEATURE_00'] == 'Minor Arterial', 'Max_speed'] = 50
        gdf.loc[gdf['FEATURE_00'] == 'Minor Arterial Ramp', 'Max_speed'] = 50
        gdf.loc[gdf['FEATURE_00'] == 'Collector', 'Max_speed'] = 50
        gdf.loc[gdf['FEATURE_00'] == 'Collector Ramp', 'Max_speed'] = 50
        gdf.loc[gdf['FEATURE_00'] == 'Local', 'Max_speed'] = 40
```

Figure 16 The code that extracts and assigns the max speed to each hierarchy of the road in Jupyter Notebook in the Anaconda environment.

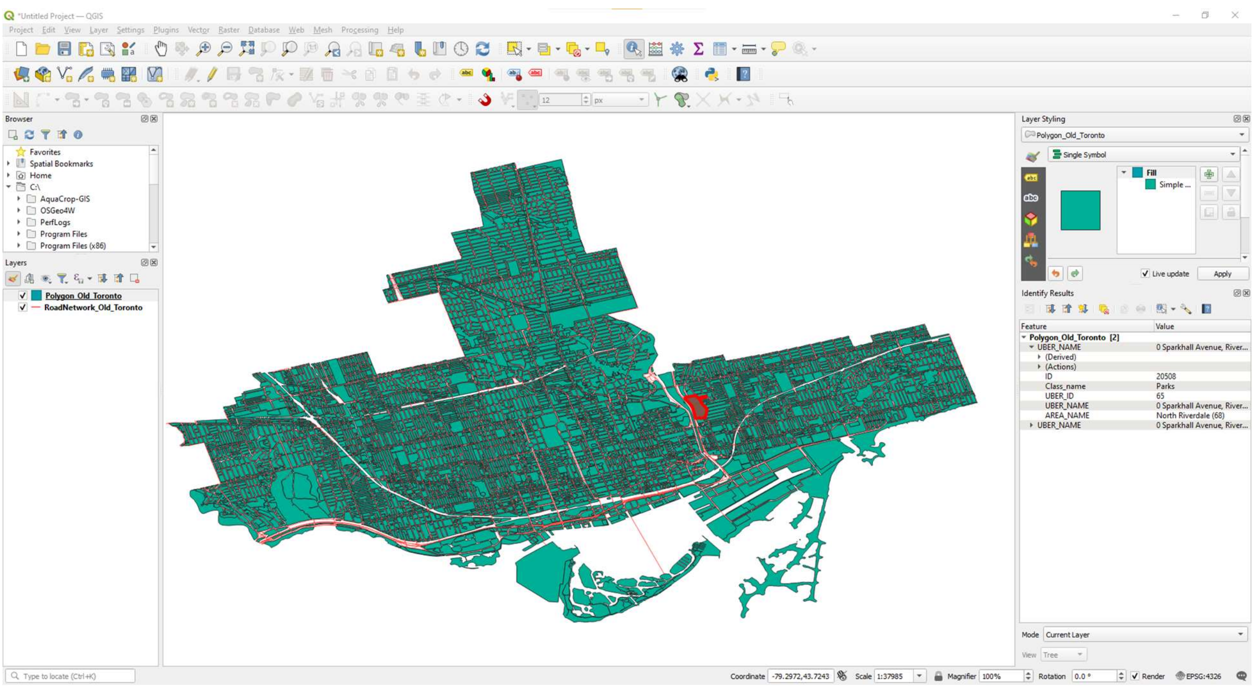

Figure 17 The final dataset that is imported into the GAMA platform

4.2 EXPERIMENT

4.2.1 COMPUTATIONAL SETTINGS

The processor of machine that was used to perform the experiments is AMD Ryzen 7 3700X eight-core processor. AMD Ryzen 7 3700X is a processor with eight cores. It is part of the third-generation Ryzen series from AMD and is based on the Zen 2 architecture. The Ryzen 7 3700X offers excellent performance for productivity tasks due to its high core count and efficient architecture. It has a base clock speed of 3.6 GHz and can boost up to 4.4 GHz. With its multi-threading capabilities, it can handle demanding tasks and applications with ease. The display adapter of this machine is NVIDIA GeForce GTX 1660. The NVIDIA GeForce GTX 1660 is a graphics card designed for gaming and multimedia purposes. It belongs to the GTX 16 series from NVIDIA and is based on the Turing architecture. The GTX 1660 offers a good balance between performance and affordability. It features 6GB of GDDR5 memory and has a base clock speed of 1530 MHz.

Updating the movement of fleet agents on a road network can be computationally demanding, especially when using real-world data in agent-based modelling (ABM) platforms. The road network data, derived from the real world, often consists of a large number of road segments and complex connectivity patterns. Each fleet agent's movement needs to be continuously updated based on the current state of the road network, including factors such as traffic conditions, speed limits, and route choices. The computational demand arises from the need to calculate and update the optimal routes for each fleet agent, considering the dynamic nature of the road network. This involves performing pathfinding algorithms, such as the A-star algorithm that is used in this model, to determine the shortest or fastest paths between locations. The complexity increases further when considering factors like congestion, traffic flow, and real-time updates of road conditions.

For example, in this simulation, the speed coefficients are periodically updated for different hierarchies of road segments to mimic real-world traffic conditions. As a result, in this case study, the run time of the simulation varies based on the fleet agent population. With a fleet agent population of 150, the simulation takes approximately 17 minutes to simulate a 24-hour period. However, when the fleet agent population is increased to 200, the run time of the simulation extends to around 20 minutes.

To address these computational challenges, GAMA platforms employ various optimization techniques and algorithms. These techniques aim to optimize the efficiency of route calculations and update processes, such as utilizing data types as graphs to represent the road network. Additionally, parallel computing or distributed computing approaches can be employed to distribute the computational workload across multiple processors or systems. Meanwhile, high-performance computing (HPC) is one of the alternative approaches for tackling computational challenges. The various approaches mentioned above need to be explored and invested in for future advancements.

4.2.2 OBJECTIVES, VARIABLES AND EVALUATION

The objective of this experiment is to assess the efficacy of various fleet size and deployment strategies in lowering operational expenses and idling time while also mitigating emissions and traffic congestion. Given the presence of a base population and limitations on searching time, it is crucial to design a dynamic fleet size that can adapt to changing demands and optimize resource allocation.

The independent variables in this experiment encompass fleet size, the base population percentage of the fleet agent, limitations on searching time, demand wait-time variations and deployment strategies. Fleet size serves as a fundamental parameter to be manipulated, allowing for a comparative analysis of different fleet configurations. The base population of the fleet agents is adjusted based on the percentage required to maintain a specific number of vehicles in service. This percentage varies according to different times of the day. Additionally, limitations on searching time are imposed to restrict the maximum idling time for each fleet agent. Demand wait-time variations, such as peak hours and regular times, simulate realistic variations in travel patterns and requirements. Moreover, deployment strategies include options such as random fixed location and dynamic allocation, which determine how fleet agents are positioned and allocated across the simulation environment.

To assess the performance and outcomes of the simulation, several dependent variables have been identified as evaluation metrics. These metrics offer insights into the effectiveness and efficiency of fleet size and deployment strategies. The following evaluation metrics are utilized:

- Unfilled Demand: Measure the number of unmet travel requests that were not serviced by the fleet. By examining the extent of unfilled demand, the efficiency of the fleet deployment strategy in meeting passenger needs can be gauged. A lower number of unfilled demands indicates a more efficient fleet deployment strategy.
- Wait Time: The average wait time experienced by people agents before being picked up by a fleet agent serves as a critical evaluation metric. It measures the responsiveness and promptness of the fleet deployment strategy in meeting travel requests. Lower wait times indicate more efficient fleet management, minimizing passenger waiting and enhancing overall customer satisfaction. The wait time is evaluated by the average wait time in rush hour and the average wait time in regular time.
- Active Fleet Agent: Monitoring the number of fleet agents actively engaged in serving travel requests at any given time provides an indication of fleet utilization. This metric

reflects the efficiency of the deployment strategy in effectively allocating and dispatching fleet agents to meet the demand. Optimizing the number of active fleet agents contributes to efficient resource utilization.

By evaluating these metrics in the context of various fleet sizes and deployment strategies, valuable insights into the strengths and limitations of each approach can be gained. This comprehensive assessment will contribute to the development of optimized fleet management strategies that achieve cost reduction, minimize idle time, reduce emissions, and alleviate traffic congestion while providing a satisfactory user experience for passengers.

4.2.3 EXPERIMENT CONFIGURATION

The in-service fleet size is designed to be dynamic, taking into account the base population of the fleet and limitations on searching time to capture the dynamics of demand. The independent variables in this experiment include fleet size, the base population of the fleet agent, limitations on searching time, customer wait-time limitation and deployment strategies. By analyzing the collected data and comparing the results based on the evaluation metrics, conclusions about the effectiveness of different fleet sizes and deployment strategies in optimizing fleet operations and meeting the specified objectives can be drawn.

The baseline experiment utilizes spatial data from Old Toronto, including its population and transportation information. According to TTS, the number of trips made by residents of Old Toronto in a 24-hour period is 1,610,100. To simplify calculations and alleviate computational load, the experiment scales down this number to 1,610, representing the total travel demand generated by Old Toronto residents in a single simulation. The temporal scale of the experiment is defined as one weekday spanning 24 hours. The experiment always starts at 0 am on a given day and ends at 0 am on the following day. In the baseline condition, the parameter of patience value is not active. The other baseline experiment settings are listed below:

- Agent Population:
    - People: The number of people agents remains constant at 1610 throughout the experiment.
    - Fleet: The number of fleet agents also remains constant at 200 throughout the experiment.
- Fleet Base:
    - AM-Mid-PM: The fleet base population is calculated as 24% of the fleet size, which means there is 24% of the fleet maintains the status of in-service during AM peak, Mid-day, and PM peak.
    - Early Morning: The fleet base population is calculated as 8% of the fleet size.
    - Evening: The fleet base is calculated as 20% of the fleet size.
- Fleet Opt:
    - Rush: The fleet searching time (idling status) limit is 1800 seconds.
    - Early Morning, Mid-day, Evening: The fleet searching time (idling status) limits are set at 450s for Early Morning, 900s for Mid-day, and 900s for Evening.
- Demand:
    - No match leave - Rush: The limit of wait time for a people agent during the rush time, who does not match with a fleet agent, is set at 180 seconds.
    - No match leave - Regular: The limit of wait time for a people agent during regular time, who does not match with a fleet agent, is set at 300 seconds.

  - Leave - Rush: The limit of wait time for a people agent during the rush time, who matches with a fleet agent, is set at 300 seconds.
  - Leave - Regular: The limit of wait time for a people agent during regular time, who matches with a fleet agent, is set at 600 seconds.
- Results:
  - Wait Time Average – Rush: The average wait time experienced by people agents during rush time.
  - Wait Time Average – Regular: The average wait time experienced by people agents during the non-rush time.
  - Unfilled Demand: The number of unmet travel requests that were not serviced by the fleet.

In the context of the experiment, there are different scenarios for both people and fleets. Regarding the people scenarios, two approaches are considered. On the other hand, the fleet scenarios encompass three distinct strategies.

- People (Matching Strategies)
  - Approach 1 involves reaching out to any fleet located within a 1/3/5 km radius of the people's location.
  - Approach 2 prioritizes fleets within a 3 km radius with “searching” objectives, followed by "not in service" fleets within the same radius, and finally includes all other fleets within a 5 km radius, excluding "commute" and "pick up" fleets.
- Fleet (Dispatching Strategies)
  - Strategies 1 focuses on fleets with a searching objective, specifically targeting hot areas.

- Strategies 2, also with a searching objective, prioritizes the neighbourhoods that have low-density of fleet agents.
- Strategies 3 adopts a strategy of equal distribution across the neighbourhoods without actively engaging in searching activities.

In summary, Strategy 1 maintains the traditional behaviour of taxi drivers in fleet operations, where part of the fleet agents roams freely in search of customers. This approach relies on the assumption that high-demand areas will naturally attract more fleet agents. On the other hand, Strategy 2 adopts a more targeted approach by allocating fleet agents with “searching” objectives to neighbourhoods with lower fleet agent density. This strategy aims to address underserved areas and improve overall customer satisfaction by reducing wait times in those neighbourhoods. In contrast, Strategy 3 takes a different approach by discarding the "searching" feature altogether. Instead, it focuses on maintaining a balanced distribution of fleet agents across different communities. The goal is to ensure equitable access to transportation services for all residents, regardless of their location. By strategically allocating fleet agents based on the area of the neighbourhood, this strategy aims to optimize service coverage and reduce disparities in service availability.

After conducting the baseline experiment, the combinations of people's approach and fleet strategy that resulted in lower customer wait times or fewer unfilled demands will be further tested under different conditions. These conditions may include smaller fleet size, a different percentage of the active fleet base population or turning on the patience value system. By testing these selected combinations in various scenarios, the researchers aim to determine their performance and effectiveness in different operational settings. This iterative process allows for the identification of optimized approaches and strategies that can potentially minimize wait times and reduce the number of unfulfilled customer demands, leading to improved overall service quality.

4.2.4 ANALYSIS AND INSIGHT

The experiment involved testing different scenarios for both people and fleets while maintaining consistent settings for agent populations, base populations of in-service fleet agents, searching time limit, and the wait time limit of people agents. Each combination of people's approach and fleet strategies is run three times. The results of these runs are presented in Table 3. The average wait time for demand in rush hours shows variation across different scenarios in Table 1Table 3. It ranges from a minimum of 60 seconds (in simulation 3) to a maximum of 122 seconds (in simulation 5). The overall average wait time for demand in rush hours is 87.6 seconds. In Figure 18, the average wait time for demand in the rush hours across the simulations ranges from a minimum of 63 minutes (Simulation 3) to a maximum of 118 minutes (Simulation 5). The average wait time generally remains relatively high, indicating potential delays for demand in rush hours in all scenarios. The average wait time for regular demand also exhibits variation among the different scenarios. It ranges from a minimum of 48 seconds to a maximum of 102 seconds.

During the simulations, it was observed that implementing stricter wait time limits resulted in a higher number of unfulfilled demands. The table above lists five simulations that utilized wait time limits of 300 seconds during rush hour and 600 seconds during regular times. If a people agent did not find a matching fleet agent within 180 seconds and 300 seconds, respectively, they would leave the system before reaching the wait time limit. While the time settings were generated based on common sense, future studies should incorporate realistic data to achieve more comprehensive and accurate results.

Results	1			2			3			4			5		
Wait-Time Average - Rush	64	69	71	74	74	70	68	62	60	117	114	116	117	115	122
Wait-Time Average - Regular	55	55	54	56	56	56	50	48	48	102	100	99	99	102	100
Unfilled Demand	3	5	7	10	11	10	6	2	6	6	4	7	10	12	16
People Approach	1	1	1	1	1	1	1	1	1	2	2	2	2	2	2
Fleet Strategy	1	1	1	2	2	2	3	3	3	1	1	1	2	2	2

Table 3 Baseline experiment: people = 1610, fleet = 200

The optimization time of fleet agents has an impact on simulations utilizing any fleet strategies except for Strategies 3. In these simulations, a shorter optimization time, indicating a more restrictive time limit on idling, resulted in better outcomes. To explore this further, a simulation was conducted using half the optimization time compared to the other scenarios. The comparison simulation run with half the optimization time resulted in fewer instances of unfilled demand and reduced customer wait times. These outcomes suggest that a shorter optimization time for fleet agents contributes to improved performance in terms of meeting customer demands and minimizing wait times.

The simulation outcomes indicate that both the people approach (matching strategies) and the fleet strategies (dispatching strategies) have an impact on the results. Simulation 1 and 2, which employed Matching Strategies 1, exhibited lower average customer wait times compared to Simulation 4 and 5. On the other hand, Simulation 1 and 4, utilizing Fleet Strategies 1, demonstrated a lower number of instances of unfilled demand. These findings suggest that the choice of matching strategies can influence customer wait times, while the selection of fleet strategies can affect the number of unfilled demands in the simulation.

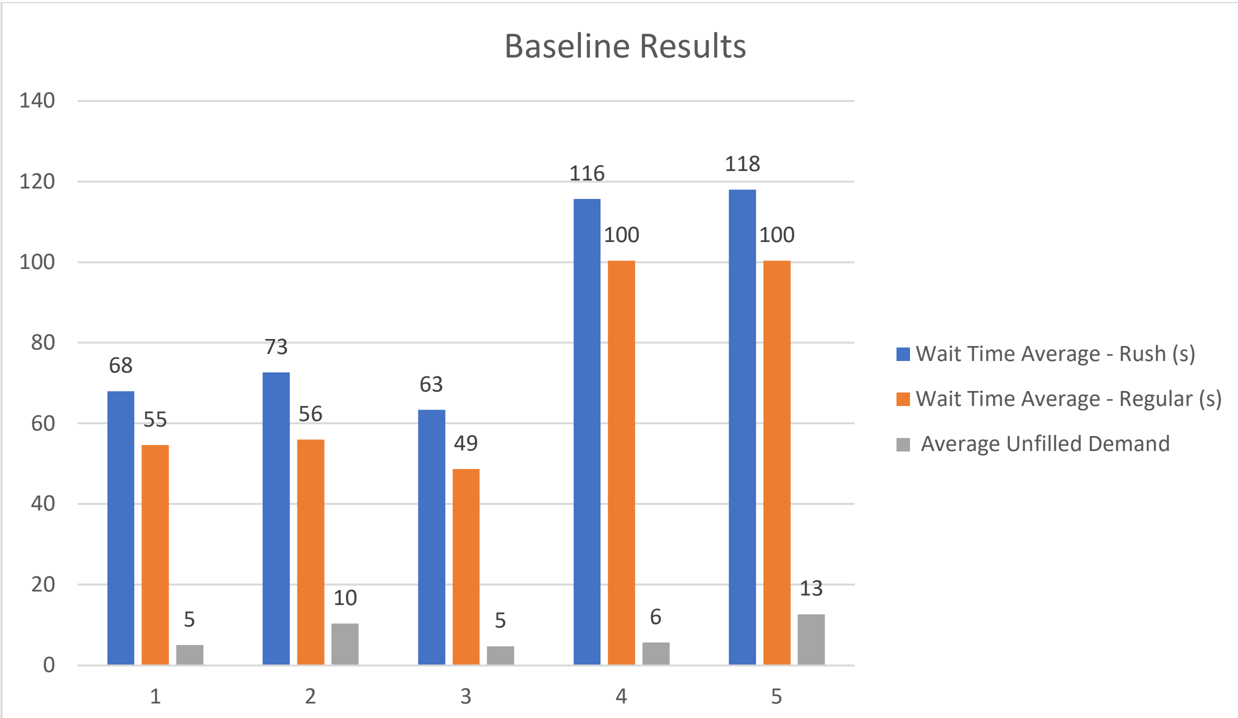

Figure 18 Baseline simulation results

Simulation 3, where people agents use Approach 1 (reaching out to any fleet located within a 1/3/5 km radius of the people's location.) and fleet agents employ Strategy 3 (a strategy of equal distribution across the neighbourhoods without actively engaging in searching activities), stood out as the most promising combination with better results. The settings of Simulation 3 are presented in Table 4. Figure 19 illustrates the dynamics of the in-service fleet, people demand and unfilled demand in one of the iterations.

	3
Step	1 min
Agent population	
People	1610
Fleet	200

Fleet Base	
AM-Mid-PM	N/A
EarlyMorning	N/A
Evening	N/A
Fleet Opt	
Rush	N/A
EarlyMorning	N/A
Mid	N/A
Evening	N/A
Demand	
No match leave - Rush	180
No match leave - NoRush	300
Leave - Rush	300
Leave - NoRush	600
Scenarios	
People Scenario	1
Fleet Scenario	3

Table 4 Experiment settings for simulation 3

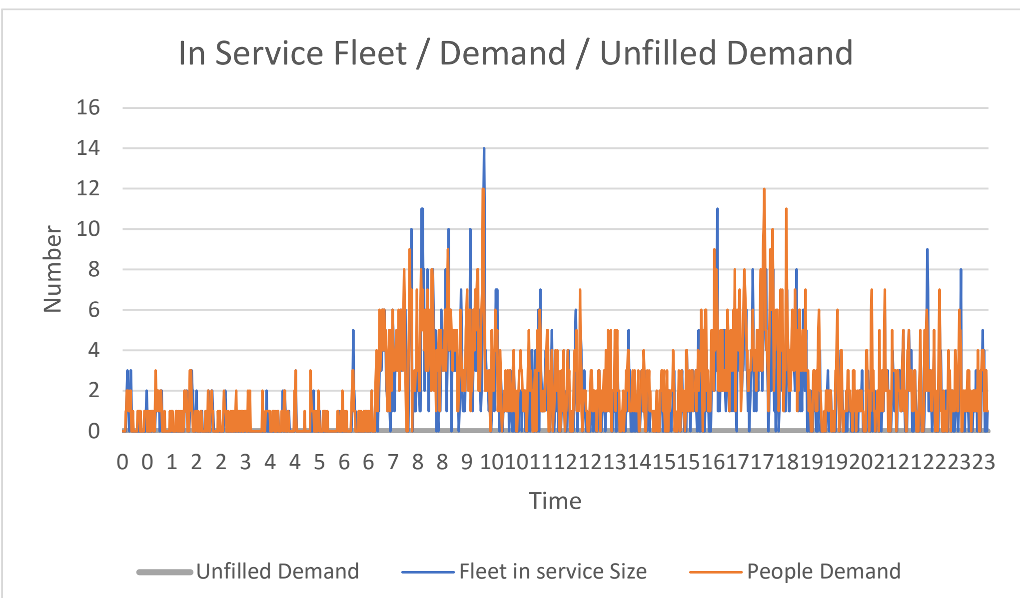

Figure 19 One iteration in simulation 3

As a result, further simulations were conducted using the same approach and strategy combination as Simulation 3 to explore the effects of different fleet sizes. In the simulation with a fleet size of 150, the results were found to be acceptable. During rush hour, the average wait time experienced by customers was 76 seconds, while it decreased to 60 seconds during regular hours. These results indicate a relatively efficient service provided by the fleet. Moreover, the number of unfilled demands, representing unmet customer requests, was recorded as 9, which is relatively low and suggests that the fleet could cater to most of the travel demands. When comparing the result from the same conditions with a fleet size of 200, it was observed that there was only a slight increase of around 10 seconds in the wait time. Figure 20 presents a column chart that illustrates the comparison between two different fleet sizes: 200 and 150. It provides a clear visualization of the impact of reducing the fleet size to 150 on the performance of the system.

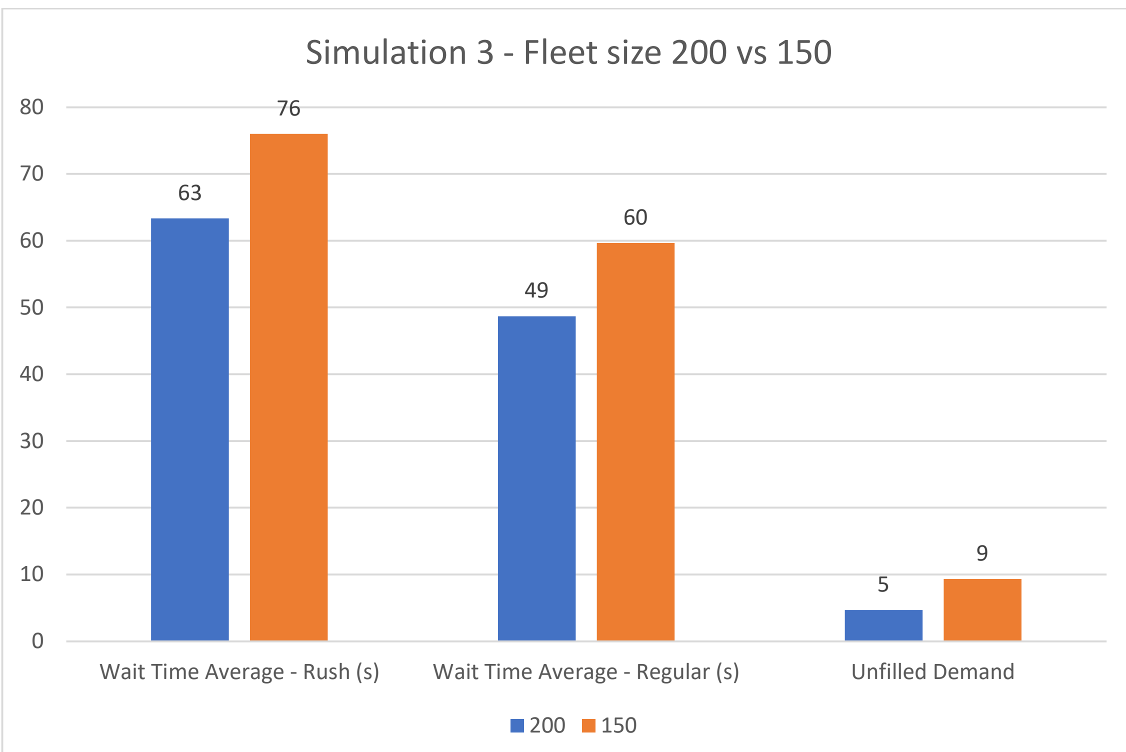

Figure 20 Simulation 3 results comparison between fleet size 200 vs 150.

However, when the fleet size was further reduced to 100, the results took a downturn and became unfavourable. During rush hour, the average wait time experienced by customers significantly increased to 125 seconds, indicating a substantial delay in obtaining transportation. Even during regular hours, the average wait time reached 77 seconds, which is still considerably high. Additionally, the number of unfilled demands surged to 41, reflecting a significant number of unattended travel requests and a decrease in service quality. These findings highlight the importance of maintaining an adequate fleet size to ensure timely service and minimize customer wait times. The optimal fleet size seems to be around 150, where the system can efficiently handle a substantial portion of the travel demands. Reducing the fleet size to 100 adversely affects service performance, leading to prolonged wait times and a significant increase in unmet travel demands.

Results	Patience		
Wait-Time Average - Rush	64	65	63
Wait-Time Average - Regular	46	51	50
Unfilled Demand	12	19	17
People Approach	1	1	1
Fleet Strategy	3	3	3

Table 5 Results from the simulations that introduced patience value

By incorporating the patience value, an element of randomness in customers' decision-making process is introduced to the system. The patience value represents the tolerance level of customers for waiting, and it influences their choices regarding whether to wait for a vehicle or seek alternative transportation options. This randomness adds a realistic aspect to the simulation, as real-life customers may exhibit varying levels of patience and make different decisions based on their individual preferences and circumstances. As illustrated in Table 5, with a fleet size of 200, the wait time during rush hour and regular time remains the same as in previous simulations' results. However, what becomes apparent is the impact of the patience value on the average unfilled demand. In this case, with the inclusion of the patient value, the average unfilled demand across the three iterations increases to 16. This is 11 higher than the average result obtained from previous simulations that did not incorporate the patience value, as is shown in Figure 21. The higher unfilled demand suggests that some customers, due to their patience level, may choose to forego waiting for a vehicle and seek alternative transportation options, resulting in a greater number of unmet demands. This highlights the influence of customer behaviour and preferences on the overall performance of the fleet system.

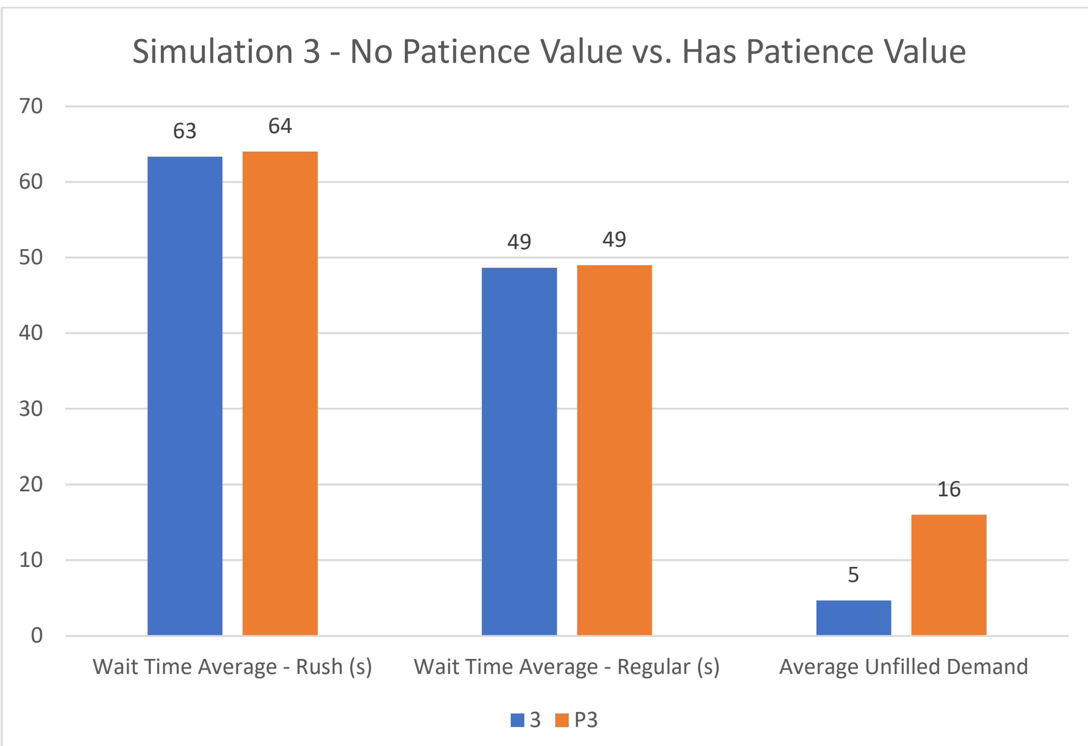

Figure 21 Comparison of the simulation without the patience value setting vs. the simulation with the patience value setting

CHAPTER 5: CONCLUSIONS AND FUTURE WORK

5.1 CONCLUSIONS AND LIMITATIONS

The rise of autonomous vehicles (AVs) fleet presents a promising solution to address traffic challenges and unlock new possibilities in the transportation industry. AV fleets operate without human drivers, whereas traditional taxi fleets rely on drivers to operate the vehicles. This driverless characteristic of AVs eliminates the need for driver wages, enables 24/7 operation, and reduces the likelihood of human errors. Additionally, AV fleets have the potential to be more cost-efficient compared to traditional taxi fleets. Through strategic deployment based on demand patterns, AVs can optimize vehicle utilization and minimize idling time. In other words, each fleet member can focus on maximizing the entire fleet's profit and customer experience. In this thesis, autonomous vehicles fleet allow for efficient deactivation and removal of idle vehicles from the system until they are required again. This operational strategy not only directly contributes to the reduction of emissions but also helps alleviate traffic congestion. Through the analysis of demand dynamics at different spatial and temporal scales, the simulation model enables the identification of size variation trends for autonomous vehicle fleets. This valuable insight empowers operators to allocate and manage resources more effectively, ensuring they align with the evolving needs of commuters. Ultimately, these efforts paved the way for a more sustainable and efficient transportation system.

The key finding of this thesis is that the strategy of equal distribution across neighbourhoods without actively engaging in searching activities showed promising results. Specifically, when the fleet size was optimized at 150 vehicles, the average wait time during rush hour was 76 seconds, and during regular hours it was 60 seconds. These results indicate that the strategy effectively managed customer wait times within the specified limits (180 seconds during rush hour and 300 seconds during regular hours) when the total demand was scaled down to 1610. This finding suggests that by distributing resources evenly across neighbourhoods, without relying on active searching activities, it is possible to achieve efficient and balanced service provision, resulting in improved customer satisfaction and reduced wait times. The findings of this thesis have several implications for the operation and management of autonomous vehicle fleets in the context of transportation services. Firstly, the strategy of equal distribution across neighbourhoods without active searching activities presents a viable approach to optimize fleet deployment. Secondly, the identified optimal fleet size of 150 vehicles offers insights into resource allocation. Operating a fleet of this size allows for efficient utilization of vehicles while meeting the demand requirements during both rush hour and regular hours. This information might guide fleet operators in determining the appropriate size of their autonomous vehicle fleets.

While this study provides valuable insights into the operation and management of autonomous vehicle fleets, it is important to acknowledge its limitations. These limitations highlight areas for further research and improvement in future studies. Firstly, the number of experiment iterations and scenarios conducted in this study was limited due to the computational resources required. Running extensive simulations with larger sample sizes and a broader range of scenarios would provide more robust and comprehensive results. However, the time-consuming nature of the computations imposed constraints on the scope of the study. Furthermore, the study focuses on a specific geographical area, namely Old Toronto, rather than examining the entire city or multiple cities. This limitation arises from the lack of sufficient computational power to simulate and analyze larger-scale urban environments. Consequently, the findings may not fully capture the complexities and dynamics of transportation systems in larger or different urban contexts. Future research should consider expanding the study to include a broader geographical scope for more representative and generalizable results. Another limitation is the lack of real-world data integration. For example, because precise spatiotemporal traffic flow data is unavailable, the speed of the fleet agent is determined solely by the maximum speed attributes associated with each road segment. In other words, the speed of the fleet agent remains unaffected by real-time traffic flow conditions. This limitation prevents the simulation from accurately reproducing real-world situations like traffic congestion during peak hours. While the study employed simulation models to mimic the behaviour of autonomous vehicle fleets and capture demand patterns, real-world data would provide more accurate and reliable insights. Incorporating actual data, such as historical travel patterns, traffic flows, and demand variations, could also enhance the validity and applicability of the findings. Despite these limitations, the study provides valuable insights into the improvement of fleet size and deployment strategies for autonomous vehicle fleets. By acknowledging these limitations and building upon the study's findings, future research can address these gaps and further advance our understanding of autonomous vehicle fleet management in more comprehensive and realistic ways.

5.2 FUTURE WORK

The conducted experiments have provided valuable insights into optimizing the performance of the transportation system by exploring different combinations of people's approaches and fleet strategies. Building on the findings mentioned previously, and there are several areas of future work that can be explored to further improve the accuracy, responsiveness, and overall quality of the transportation multi-agent model. First of all, the neighbourhood layer is able to involve more attributes. More attributes, such as demographic information and historical travel data, enable the design of fleet distribution, dispatch, and searching algorithms that take into account multiple factors. It enables the simulation to incorporate more intelligent and context-aware decision-making processes. Secondly, there is one promising avenue for future work is the integration of shared mobility services into this transportation simulation model. Shared mobility refers to the concept of sharing transportation resources among multiple users. By incorporating shared mobility options, the efficiency and sustainability of transportation systems can be enhanced while providing convenient and cost-effective travel solutions. There are several aspects that can be explored. Firstly, research can delve into the development and testing of more sophisticated dispatching algorithms that leverage real-time data and optimization techniques. These algorithms can take into account various factors such as passenger demand, traffic conditions, vehicle availability, and time of the day to make intelligent dispatching decisions. Furthermore, developing intelligent matching algorithms and platforms that facilitate the seamless matching of passengers with available shared vehicles can greatly improve the utilization of shared mobility services. These algorithms can consider factors such as passenger preferences, trip routes, and time constraints to optimize matching and enhance user experience. Additionally, integrating shared mobility with existing public transit networks in the simulation can create a comprehensive and interconnected transportation ecosystem. These approaches allow modellers to obtain a comprehensive understanding of the entire transportation system, considering the interconnectivity of various modes. As a result, the outcomes of these modelling efforts can provide valuable insights for transportation planners, empowering them to make more informed and effective decisions when devising strategies. Another area of future work with substantial potential is the integration of advanced demand prediction techniques into agent-based modelling. By incorporating sophisticated algorithms and data analytics, the model can accurately forecast travel demand patterns and trends, taking into account various factors such as time, weather conditions, events, and user behaviour. This integration would enhance the predictive capabilities of the model, enabling better anticipation of transportation needs and facilitating more efficient resource allocation and planning. Moreover, by continuously updating and refining the demand prediction models based on real-time data, the simulation can adapt to dynamic changes in travel patterns, ensuring its relevance and effectiveness in guiding decision-making processes for transportation planners.

REFERENCES

Balbi, S., & Giupponi, C. (2009). Reviewing Agent-Based Modelling of Socio-Ecosystems: A Methodology for the Analysis of Climate Change Adaptation and Sustainability. *SSRN Electronic Journal*. https://doi.org/10.2139/ssrn.1457625

Bazzan, A. L. C., & Klügl, F. (2014). A review on agent-based technology for traffic and transportation. *The Knowledge Engineering Review*, *29*(3), 375–403. https://doi.org/10.1017/S0269888913000118

Boesch, P. M., Ciari, F., & Axhausen, K. W. (2016). Autonomous Vehicle Fleet Sizes Required to Serve Different Levels of Demand. *Transportation Research Record: Journal of the Transportation Research Board*, *2542*(1), 111–119. https://doi.org/10.3141/2542-13

Bonabeau, E. (2002). Agent-based modeling: Methods and techniques for simulating human systems. *Proceedings of the National Academy of Sciences*, *99*(suppl_3), 7280–7287. https://doi.org/10.1073/pnas.082080899

Bösch, P. M., Becker, F., Becker, H., & Axhausen, K. W. (2018). Cost-based analysis of autonomous mobility services. *Transport Policy*, *64*, 76–91. https://doi.org/10.1016/j.tranpol.2017.09.005

Caillou, P., Gaudou, B., Grignard, A., Truong, C. Q., & Taillandier, P. (2017). A Simple-to-Use BDI Architecture for Agent-Based Modeling and Simulation. In W. Jager, R. Verbrugge, A. Flache, G. De Roo, L. Hoogduin, & C. Hemelrijk (Eds.), *Advances in Social Simulation 2015* (Vol. 528, pp. 15–28). Springer International Publishing. https://doi.org/10.1007/978-3-319-47253-9_2

Chen, T. D., Kockelman, K. M., & Hanna, J. P. (2016). Operations of a shared, autonomous, electric vehicle fleet: Implications of vehicle & charging infrastructure decisions.

Transportation Research Part A: Policy and Practice, *94*, 243–254. https://doi.org/10.1016/j.tra.2016.08.020

Chunlin He, He Xiao, Wen Dong, & Liping Deng. (2010). Dynamic group behavior for real-time multi-agent crowd simulation. *2010 The 2nd International Conference on Computer and Automation Engineering (ICCAE)*, 544–546. https://doi.org/10.1109/ICCAE.2010.5451876

City of Toronto. (n.d.). *City of Toronto Road Classification System*.

Crooks, A. (2018). *Agent-based modelling and geographical information systems: A practical primer* (1st edition). SAGE Publications.

Cuevas, E. (2020). An agent-based model to evaluate the COVID-19 transmission risks in facilities. *Computers in Biology and Medicine*, *121*, 103827. https://doi.org/10.1016/j.compbiomed.2020.103827

Davies, B., Romanowska, I., Harris, K., & Crabtree, S. A. (2019). Combining Geographic Information Systems and Agent-Based Models in Archaeology: Part 2 of 3. *Advances in Archaeological Practice*, *7*(2), 185–193. https://doi.org/10.1017/aap.2019.5

Fortin, M. (2022). *Toronto Land Use Spatial Data—Parcel-level—(2019-2021)* [dataset]. Borealis. https://doi.org/10.5683/SP3/1VMJAG

Grimm, V., Berger, U., Bastiansen, F., Eliassen, S., Ginot, V., Giske, J., Goss-Custard, J., Grand, T., Heinz, S. K., Huse, G., Huth, A., Jepsen, J. U., Jørgensen, C., Mooij, W. M., Müller, B., Pe'er, G., Piou, C., Railsback, S. F., Robbins, A. M., … DeAngelis, D. L. (2006). A standard protocol for describing individual-based and agent-based models. *Ecological Modelling*, *198*(1–2), 115–126. https://doi.org/10.1016/j.ecolmodel.2006.04.023

Guo, D., Ren, B., & Wang, C. (2008). Integrated Agent-Based Modeling with GIS for Large Scale Emergency Simulation. In L. Kang, Z. Cai, X. Yan, & Y. Liu (Eds.), *Advances in Computation and Intelligence* (Vol. 5370, pp. 618–625). Springer Berlin Heidelberg. https://doi.org/10.1007/978-3-540-92137-0_68

Han, Z., Zhang, K., Yin, H., & Zhu, Y. (2015). An urban traffic simulation system based on multi-agent modeling. *The 27th Chinese Control and Decision Conference (2015 CCDC)*, 6378–6383. https://doi.org/10.1109/CCDC.2015.7161966

Hart, P., Nilsson, N., & Raphael, B. (1968). A Formal Basis for the Heuristic Determination of Minimum Cost Paths. *IEEE Transactions on Systems Science and Cybernetics*, *4*(2), 100–107. https://doi.org/10.1109/TSSC.1968.300136

Heinrichs, M., Krajzewicz, D., Cyganski, R., & Von Schmidt, A. (2017). Introduction of car sharing into existing car fleets in microscopic travel demand modelling. *Personal and Ubiquitous Computing*, *21*(6), 1055–1065. https://doi.org/10.1007/s00779-017-1031-3

Heppenstall, A., Evans, A., & Birkin, M. (2006). Using Hybrid Agent-Based Systems to Model Spatially-Influenced Retail Markets.pdf. *Journal of Artificial Societies and Social Simulation*, *9*(3), 2.

Jennings, N. R. (2000). On agent-based software engineering. *Artificial Intelligence*, *117*(2), 277–296. https://doi.org/10.1016/S0004-3702(99)00107-1

Jing, P., Hu, H., Zhan, F., Chen, Y., & Shi, Y. (2020). Agent-Based Simulation of Autonomous Vehicles: A Systematic Literature Review. *IEEE Access*, *8*, 79089–79103. https://doi.org/10.1109/ACCESS.2020.2990295

Kagho, G. O., Balac, M., & Axhausen, K. W. (2020). Agent-Based Models in Transport Planning: Current State, Issues, and Expectations. *Procedia Computer Science*, *170*, 726–732. https://doi.org/10.1016/j.procs.2020.03.164

Kloppel, M., Schmid, W., & Lienkamp, M. (2019). Agent-based Simulation of a Car-sharing System with Hydrogen-powered Vehicles. *2019 Fourteenth International Conference on Ecological Vehicles and Renewable Energies (EVER)*, 1–8. https://doi.org/10.1109/EVER.2019.8813666

Leombruni, R., Richiardi, M., Saam, N. J., & Sonnessa, M. (n.d.). *A Common Protocol for Agent-Based Social Simulation*.

Loeb, B., Kockelman, K. M., & Liu, J. (2018). Shared autonomous electric vehicle (SAEV) operations across the Austin, Texas network with charging infrastructure decisions. *Transportation Research Part C: Emerging Technologies*, *89*, 222–233. https://doi.org/10.1016/j.trc.2018.01.019

Lu, M., Taiebat, M., Xu, M., & Hsu, S.-C. (2018). Multiagent Spatial Simulation of Autonomous Taxis for Urban Commute: Travel Economics and Environmental Impacts. *Journal of Urban Planning and Development*, *144*(4), 04018033. https://doi.org/10.1061/(ASCE)UP.1943-5444.0000469

Macal, C., & North, M. (n.d.). *INTRODUCTORY TUTORIAL: AGENT-BASED MODELING AND SIMULATION*. 15.

Maciejewski, M., & Bischoff, J. (2018). CONGESTION EFFECTS OF AUTONOMOUS TAXI FLEETS. *Transport*, *33*(4), 971–980. https://doi.org/10.3846/16484142.2017.1347827

Marley, J., Hyde, A., Salkeld, J. H., Prima, M.-C., Parrott, L., Senger, S. E., & Tyson, R. C. (2017). Does human education reduce conflicts between humans and bears? An agent-based modelling approach. *Ecological Modelling*, *343*, 15–24. https://doi.org/10.1016/j.ecolmodel.2016.10.013

Martinez, L. M., Correia, G. H. A., & Viegas, J. M. (2015). An agent-based simulation model to assess the impacts of introducing a shared-taxi system: An application to Lisbon (Portugal): AN APPLICATION TO LISBON (PORTUGAL). *Journal of Advanced Transportation*, *49*(3), 475–495. https://doi.org/10.1002/atr.1283

Martinez, L. M., & Viegas, J. M. (2017). Assessing the impacts of deploying a shared self-driving urban mobility system: An agent-based model applied to the city of Lisbon, Portugal. *International Journal of Transportation Science and Technology*, *6*(1), 13–27. https://doi.org/10.1016/j.ijtst.2017.05.005

McNally, M. G. (n.d.). *The Four Step Model*.

Negahban, A., & Yilmaz, L. (2014). Agent-based simulation applications in marketing research: An integrated review. *Journal of Simulation*, *8*(2), 129–142. https://doi.org/10.1057/jos.2013.21

Singh, D., Padgham, L., & Logan, B. (2016). Integrating BDI Agents with Agent-Based Simulation Platforms. *Autonomous Agents and Multi-Agent Systems*, *30*(6), 1050–1071. https://doi.org/10.1007/s10458-016-9332-x

Taillandier, P., Bourgais, M., Caillou, P., Adam, C., & Gaudou, B. (2017). A BDI Agent Architecture for the GAMA Modeling and Simulation Platform. In L. G. Nardin & L. Antunes (Eds.), *Multi-Agent Based Simulation XVII* (Vol. 10399, pp. 3–23). Springer International Publishing. https://doi.org/10.1007/978-3-319-67477-3_1

Taillandier, P., Gaudou, B., Grignard, A., Huynh, Q.-N., Marilleau, N., Caillou, P., Philippon, D., & Drogoul, A. (2019). Building, composing and experimenting complex spatial models with the GAMA platform. *GeoInformatica*, *23*(2), 299–322. https://doi.org/10.1007/s10707-018-00339-6

Taillandier, P., Therond, O., & Gaudou, B. (n.d.). *A new BDI agent architecture based on the belief theory. Application to the modelling of cropping plan decision-making*.

Thompson, J. R., Frezza, D., Necioglu, B., Cohen, M. L., Hoffman, K., & Rosfjord, K. (2019). Interdependent Critical Infrastructure Model (ICIM): An agent-based model of power and water infrastructure. *International Journal of Critical Infrastructure Protection*, *24*, 144–165. https://doi.org/10.1016/j.ijcip.2018.12.002

Willoughby, J. R., & Christie, M. R. (2019). Long-term demographic and genetic effects of releasing captive-born individuals into the wild. *Conservation Biology*, *33*(2), 377–388. https://doi.org/10.1111/cobi.13217

Yin, W., Murray-Tuite, P., Ukkusuri, S. V., & Gladwin, H. (2014). An agent-based modeling system for travel demand simulation for hurricane evacuation. *Transportation Research Part C: Emerging Technologies*, *42*, 44–59. https://doi.org/10.1016/j.trc.2014.02.015

Zhang, L., Wang, J., & Shi, Q. (2014). Multi-agent based modeling and simulating for evacuation process in stadium. *Journal of Systems Science and Complexity*, *27*(3), 430–444. https://doi.org/10.1007/s11424-014-3029-5

Zhao, B., Kumar, K., Casey, G., & Soga, K. (2019). Agent-Based Model (ABM) for City-Scale Traffic Simulation: A Case Study on San Francisco. *International Conference on Smart Infrastructure and Construction 2019 (ICSIC)*, 203–212. https://doi.org/10.1680/icsic.64669.203

Zou, M., Li, M., Lin, X., Xiong, C., Mao, C., Wan, C., Zhang, K., & Yu, J. (2016). An agent-based choice model for travel mode and departure time and its case study in Beijing.

Transportation Research Part C: Emerging Technologies, *64*, 133–147. https://doi.org/10.1016/j.trc.2015.06.006

APPENDIX

State Variables – Neighbourhood:

Attribute	Description
area_name	The name of the area or neighbourhood
uber_name	The name of the neighbourhood in the Uber system
move_id	The ID associated with the neighbourhood
area_sqm	The size of the neighbourhood in square meters
area_sqkm	The size of the neighbourhood in square kilometers (derived from area_sqm)
neighbourhood_demand_num	The number of people inside the neighbourhood making requests
neighbourhood_potential_num	The number of people inside the neighbourhood not making requests
neighbourhood_supply_num	The number of vehicles in the fleet that are not in service within the neighbourhood
neighbourhood_supply_sqkm_num	The number of vehicles per square kilometer of the neighbourhood (derived from neighbourhood_supply_num and area_sqkm)

State Variables – Building:

Attribute	Description

type	The type or category of the building
area_name	The name of the area where the building is located
move_id	The ID associated with the building
color	The color of the building in RGB format

State Variables – Road:

Attribute	**Description**
type	The type or category of the road skills

State Variables – People:

Attribute	**Description**
request_time_hour	The hour at which the request is made
request_time_min	The minute at which the request is made
request	Indicates whether a request has been made (initially set to false)
initial_time	The initial request time for the person
target_building	The building that the person intends to reach
purpose	The purpose of the person's travel
objective	The objective of the person's travel
area_update	The updated geometry of the area where the person is located
area_name	The name of the area where the person is located
move_id	The ID associated with the person's location

current_fleet	The current fleet the person is associated with
patience_val	The initial level of patience for the person
patience_val_loss	The amount of patience lost over time
patience_lose	Indicates whether the person has lost patience
wait_time	The time the person has waited
leave_trig	Indicates whether the person triggers leaving
no_wait_trig	Indicates whether the person triggers no waiting
wait_for_pickup	Indicates whether the person is waiting for pickup
wait_dist	The distance the person is willing to wait for pickup (default: 1.00 km)

State Variables – Fleet:

Attribute	Description
area_update_neighbour	A list of updated geometries of neighboring areas
area_neighbour_union	The union of geometries of neighboring areas
target_building	The building the fleet is targeting
objective	The objective of the fleet's movement
current_ppl	The person currently associated with the fleet
check_access	Indicates whether access is being checked
target	The target point the fleet is moving towards
target_ppl	The target point associated with a person in the fleet
target_des	The target point associated with a destination in the fleet

dep_time	The departure time of the fleet
arr_time	The arrival time of the fleet
total_time	The total time taken by the fleet for the movement
searching_timer	The timer used for searching
start_searching_time	The time when the searching starts
end_searching_time	The time when the searching ends
view_dist	The viewing distance of the fleet