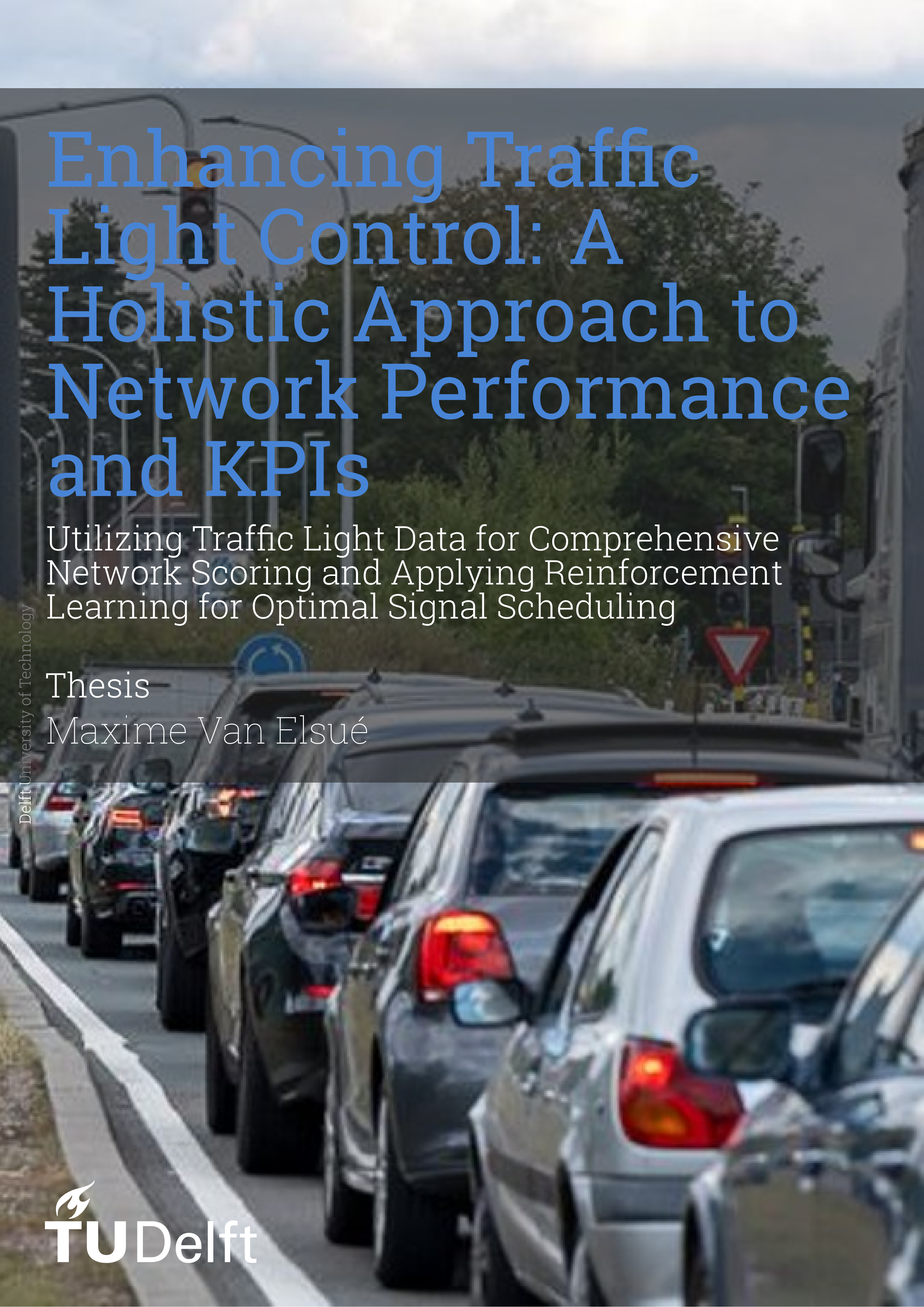


Enhancing Traffic Light Control: A Holistic Approach to Network Performance and KPIs

Utilizing Traffic Light Data for Comprehensive Network Scoring and Applying Reinforcement Learning for Optimal Signal Scheduling

Thesis

Maxime Van Elsué

Delft University of Technology

TU Delft

Enhancing Traffic Light Control: A Holistic Approach to Network Performance and KPIs

Utilizing Traffic Light Data for Comprehensive Network Scoring and Applying Reinforcement Learning for Optimal Signal Scheduling

by

Maxime Van Elsué

Van Elsué
Maxime

Chair:	Dr.ir. A.J. (Adam) Pel
University supervisors:	Dr.ir. A.M. (Maria) Salomons
University supervisors:	J. (Rina) Cheng
Companny supervisor:	ir. M. (Mark) Sloot
Project Duration:	November, 2024 - February, 2025
Faculty:	Faculty of Civil Engineering, Delft

Cover: (DPG Media Privacy Gate, n.d.)
Style: TU Delft Report Style, with modifications by Daan Zwaneveld

TU Delft

Summary

Cities worldwide face significant congestion issues, leading to time wasted in traffic, increased environmental impact due to idling engines, and diminished quality of life due to noise and safety concerns. Addressing these problems necessitates proactive policy changes, such as implementing low-carbon zones and optimizing traffic light schedules. This research investigates how traffic lights can be leveraged to alleviate congestion while also enhancing air quality, equity, and overall urban sustainability. By integrating policy changes into traffic light schedules, this study aims to provide actionable insights for municipalities and improve day-to-day operations for traffic managers, urban planners, and policymakers.

Despite extensive research into traffic management technologies, current studies often focus predominantly on car traffic flow and waiting times, occasionally overlooking broader urban mobility aspects, including cyclists, pedestrians, and public transport. Furthermore, factors like environmental pollution and noise are gaining importance but are still not sufficiently integrated into traffic management systems. This research addresses this gap by proposing a more comprehensive approach that includes these diverse factors and stakeholders, aiming to contribute to a more liveable and efficient urban traffic network.

Notably, this study introduces a novel two-level Reinforcement Learning (RL) framework using Deep Q-Networks (DQNs) to address traffic management. Unlike traditional systems, the two-level RL DQN approach provides dynamic coordination between local traffic lights and a central network-wide management system. This holistic methodology is designed to optimize immediate traffic flow while simultaneously supporting long-term policy goals concerning sustainability, equity, and adaptability. This research introduces the concept of a ”global observer” integrated with local agents, a feature that ensures network-wide policy alignment by periodically updating traffic light parameters based on broader societal goals. This novel structuring allows for detailed consideration of emissions, noise, and other externalities, thus setting a new standard for urban mobility management.

The primary goal of this research is to demonstrate how integrating reinforcement learning algorithms can optimize traffic light control to support broader societal priorities such as reducing congestion, minimizing emissions, and promoting equitable traffic management. To achieve this, the research aims to identify key performance indicators (KPIs) that reflect policy goals and examine how reinforcement learning can be applied both in real time and for long-term urban planning. Specific tasks include measuring these KPIs within simulations, analyzing network-level performance to pinpoint bottlenecks, and evaluating the effectiveness of RL-based traffic light controllers under real-time demand fluctuations and long-term policy scenarios. The study seeks to answer questions related to optimal metric calculations, intersection performance aggregation, and the application of RL for sustainable and efficient traffic system management.

Main Research Questions

- What are KPIs for evaluating policy goals, and how can these metrics reflect priorities such as reducing congestion, minimizing emissions, and promoting equitable outcomes?
- How can reinforcement learning algorithms be applied to optimize real-time traffic light control in a way that supports policy goals such as equitable traffic management, sustainable urban mobility, and efficient traffic flow?

This study used a mixed-method approach that combined qualitative insights with quantitative data to ensure a comprehensive analysis of traffic management through reinforcement learning (RL). Data collection was carried out through simulations using the traffic modeling tool PTV Vissim (software for micro simulation), supplemented by case studies in Almelo, The Netherlands. Quantifiable metrics included vehicle delays, queue lengths, emissions, and performance. In addition, municipal policies and urban planning documents provided a qualitative context, ensuring that the RL models were in line with broader policy goals. In PTV Vissim, a digital twin of Almelo's traffic environment was created, including virtual vehicles, intersections and road segments. Scenario-based experiments enabled the assessment of control strategies across varying traffic conditions and policy priorities. Data analysis involved statistical evaluations of KPIs and performance indicators, combined with reinforcement learning modeling using Deep Q-Networks (DQNs). Graph theory was applied to assign weighted importance to different road segments, and iterative visualizations on a custom dashboard helped translate quantitative results into actionable policy insights. This reinforcing framework ensured that the RL algorithms dynamically refined traffic light control parameters for optimized urban mobility and sustainability. This research successfully demonstrated how integrating reinforcement learning (RL) with graph theory can enhance urban traffic management. Using Python and PTV Vissim, a prototype dashboard is developed to visualize network performance and KPI scores, allowing municipalities to align RL models with their policy goals. This holistic approach highlights the importance of balancing car performance and broader objectives such as emissions and safety.

Reinforcement Learning (RL), using Deep Q-Networks (DQNs), showed promising results, reducing waiting times by 20-35 percent and emissions by over 10 percent. Local agents individually trained demonstrated that learning speed is closely related to signal group complexity, necessitating carefully normalized reward functions and optimal discount factors (between 0.8-0.85) for effective training. Adjusting these factors was crucial to prevent issues such as hidden queues and to prioritize dominant traffic flows effectively. Additionally, this study advances reinforcement learning by incorporating intricate weighting through graph theory, providing a detailed method for identifying and addressing network bottlenecks. This integration ensures that traffic light control is dynamically refined, aligning real-time results with longer-term policy goals and societal benefits.

The involvement of a global agent proved essential for network-wide optimization, considering the broader picture ignored by local agents focused on reducing waiting times. While the local agents improved performance, scenarios involving the global agent showed further gains of 0.03 to 0.2 in overall network efficiency. These gains will only grow bigger on a larger scale.

Despite the promising results, scalability and computation times pose challenges for real-time applications. Future research should address these limitations by exploring hierarchical learning frameworks and optimizing reward structures for complex intersections. Graph theory successfully identified network bottlenecks, though it struggled with capturing complex traffic dynamics such as weaving sections.

In conclusion, while RL demonstrates substantial improvements in traffic management, achieving faster and scalable RL solutions for larger networks remains a key objective. The potential of RL combined with graph theory and a visual framework for policy alignment offers valuable insights into optimizing urban mobility. Further refinement of techniques and tools can unlock larger gains, benefiting cities worldwide. These findings highlight the practical value of AI-based traffic control systems in supporting smarter urban planning and more effective mobility solutions.

Acknowledgments

I would like to sincerely thank all my supervisors for their contributions and support throughout my thesis.

First and foremost, I would like to thank Mark Sloot from Sweco for answering all my questions and guiding my project in the right direction. I also wish to thank Sweco for providing various tools that were instrumental in making this thesis a success. Furthermore, I would like to express my gratitude to Arie Schreuders and Merel Rozier from Sweco for their help in defining the thesis topic, and to Winston Ekkel from the municipality of Almelo for a guided tour by bike, visiting the intersections in the network, and for pointing out the critical issues within the network.

I would also like to thank Jingyi Cheng for her assistance in developing the reinforcement learning model and applying AI techniques. Her tips and tricks significantly contributed to achieving the desired results with the model.

Finally, I would like to thank Maria Salomons and Adam van Pel for maintaining an overview and ensuring that the report effectively communicated the core ideas. Maria's advice on traffic control strategies was particularly helpful in the development of the neural network.

Lastly, I am grateful to my family for their love, understanding, and constant motivation throughout this journey.

Contents

1 Introduction

1.1. Problem statement

Cities around the world are struggling with the effects of congestion. In addition to the time wasted waiting in traffic, these traffic jams contribute to significant damage to the environment due to vehicle engines that idle without movement [1]. Traffic jams also reduce the quality of life of residents, contributing to noise pollution and safety concerns [2]. Therefore, it is crucial to make the streets more liveable by reducing traffic. In addition, implementing policy changes, such as low-carbon zones [3], can help achieve the goal of creating more pleasant cities. This research will focus on how traffic lights can improve traffic congestion and other factors such as air quality, equity, etc. Therefore, this research will give insight into how policy changes can be integrated into the traffic light schedule to improve our cities in the future; the focus on equity, sustainability, and efficiency reflects broader societal priorities. This tool will help municipalities in their day-to-day operations. The primary users of this dashboard will be traffic managers, urban planners, and long-term policy makers. For traffic managers, the tool will enable real-time monitoring and intervention during daily operations or special events. Urban planners and policymakers will use the dashboard to simulate and evaluate long-term policy impacts, such as noise reduction strategies or equitable traffic distribution. The dashboard aims to provide actionable insights, such as identifying persistent bottlenecks, predicting congestion patterns, and testing the effects of alternative policies in a virtual environment. Its functionality will range from real-time data visualization to scenario-based forecasting, ensuring accuracy and relevance for current conditions and future predictions.

This research is carried out as part of an internship at Sweco, an urban planning and engineering consultancy. The internship provides a practical lens through which this research is conducted, ensuring that the outcomes are aligned with the real-world challenges faced by municipalities. Sweco's role in the DAES (Doorstroming als een service / Throughput as a Service) project, together with the cities of Almelo, Enschede, and partners like Heijmans, is centered on improving both traffic conditions and the quality of life for residents. Gaining insight into network performance and understanding the impact of short- and long-term measures is essential to make this project a success. Improving throughput by optimizing only traffic lights could be a very cost-effective method to improve congestion and the lives of the inhabitants of these cities.

Reinforcement learning can help establish a traffic light program that improves not only current traffic flow but also supports broader goals related to long-term planning, such as sustainability, equity, and adaptability to sudden demand fluctuations. Although both short-term and long-term goals can be approached separately, there is a significant advantage in considering traffic light control as part of the broader performance of the network rather than in isolation.

1.2. Limitation of current studies

A substantial body of research has already explored how such emerging technologies can enhance traffic management. However, ”improvement” itself is a broad and often ambiguous term. Much of the literature tends to focus on metrics such as increased traffic flow or reduced waiting times [4][5][6]. While efficient car traffic is undoubtedly vital for any city, there is a growing shift toward designing more liveable urban environments. This involves not just optimizing for cars, but also prioritizing cyclists, pedestrians, and residents. The current tools often fail to give this other mode enough attention.

In addition, factors such as noise and environmental pollution, along with the integration of public transport, are gaining importance in modern traffic management. Some studies have explored the use of reinforcement learning to create more equitable traffic systems to balance travel times between users [7]. However, these approaches often remain primarily focused on the car. This research seeks to broaden that scope by addressing the multidimensional nature of urban mobility. Considering a wider range of factors and stakeholders, including pedestrians, cyclists, and public transport, this more comprehensive approach aims to contribute to a traffic network that performs better overall.

1.3. Research objectives and questions

The goal is to find a way to make the performance of the network understandable. This is the objective of this research, not only for long-term planners but also for traffic managers who have to implement measures ad hoc.

This research is structured around a set of key questions that guide its objectives, addressing both broad goals and specific areas of interest. The first question is needed to be able to solve the second question in an holistic approach.

Main research questions

- What are KPIs for evaluating policy goals, and how can these metrics reflect priorities such as reducing congestion, minimizing emissions, and promoting equitable outcomes?
- How can reinforcement learning algorithms be applied to optimize real-time traffic light control in a way that supports policy goals such as equitable traffic management, sustainable urban mobility, and efficient traffic flow?

Sub-questions

1. **Performance Indicators**
    - How to calculate/measure these KPIs (that are used for evaluating policy goals) in the simulation (and in practice)?
2. **Network-Level Analysis**
    - How can individual intersection performance scores be combined into a single interpretable and transparent metric that reflects overall network efficiency and traffic flow patterns?
    - Which intersections can be identified as bottlenecks using graph-theoretical methods under specific traffic conditions or demand patterns, and to what extent do these bottlenecks represent the initial points of failure in the traffic network when subjected to increasing demand or disruption?
3. **Optimization and Policy Recommendations**
    - **Short term:** How effective are reinforcement learning-based traffic light controller in absorbing real-time demand fluctuations in traffic and how will it improve traffic speed and reduce delays at the intersection level?
    - **Long term:** How can reinforcement learning-based simulations be used to optimize traffic light scheduling in support of long-term policy goals?

1.4. Scope

This research offers a dual contribution to the field of intelligent traffic management. First, it develops a practical tool that provides clear and actionable insights into the performance of intersections and traffic networks. Second, it uses this performance assessment to optimize traffic light schedules through reinforcement learning, with the aim of creating systems that adapt dynamically to real-time traffic conditions. By combining network analysis with intelligent control, the research provides a foundation for smarter and more responsive traffic systems that can help municipalities achieve their mobility goals more effectively.

The scope of this research is two-fold. On the one hand, it aims to provide a tool that offers insight into the performance of intersections and networks. On the other hand, it will utilize this performance score to improve traffic light schedules using reinforcement learning. This study does not account for changes in modes of transport between individuals. This extra complexity will be important for long-term planning, but adds to the number of variables already in the system. Further research should implement mode choice to have a model that better represents reality and the complexity of trip chains and mode choice. Quite a lot of research has been conducted on OD-calibration, and this will be essential to have an adaptive and responsive simulation. The Appendix A builds on this research to show how a digital twin would be created by having a realistic OD-matrix that is updated with current demand fluctuations. When the real world sees traffic patterns change, the model can adjust its OD-matrix and will be able to fine-tune the reinforcement learning agents with the more accurate traffic conditions. The responsive model helps traffic managers make ad hoc decisions if they see something go wrong. The model will be able to foresee problems before they happen, which will give the traffic manager some more time to respond (this will depend on size of model, computing power, etc.).

The study limits itself to aspects that can be measured from a simulation. For instance, red-light violations can be estimated from a simulation but fall outside of the scope of this research and, therefore, can be extracted from real-world data. The latter can and should be studied in further research that builds on top of this research. This presents the challenge that these factors may not change when adjustments are made in the simulation. More research will be necessary to explore how these key performance indicators can be effectively transferred to a simulation. The model will include them to emphasize their importance in a holistic approach.

1.5. Significance of the study

This research has significant social implications if it demonstrates that an integrated approach, one that considers both network-wide performance and local intersection conditions, can better support key policy goals (such as equity, sustainability, and efficiency) compared to isolated control strategies, even if trade-offs between these goals are involved. This paradigm shift can transform traffic management, urging policy makers and urban planners to adopt a holistic view that balances the overall performance of the transportation network with localized issues such as emissions, traffic flow, and pedestrian safety.

The proposed dashboard serves as both a strategic and operational tool, informing long-term investment decisions and addressing immediate challenges. The tool should be coupled to outside sensors so that the simulation and the reality match. The simulation can then predict what will happen and can inform real-time traffic managers when problems arise. Traffic managers can also artificially boost certain OD-pairs, say, for example, for a football game, many people will drive to and from the stadium. The impact of this influx of people could be simulated beforehand to predict the measures that will be most effective. Integrating reinforcement learning into this approach further enhances its impact. Unlike traditional systems that focus solely on traffic metrics, the RL framework considers the quality of life for all road users, including pedestrians and cyclists. By analyzing traffic patterns, the RL model adapts to traffic demand, ensuring flexibility during events such as football matches. Many municipalities still look mainly at cars. This dashboard approach with sliders helps municipalities align their goals and policies. They can also see what the impact of certain policies is and what the impact will be on their network.

A key advantage of this method is its ability to achieve unattainable optimization levels for humans. RL-driven adaptive traffic light systems can improve traffic flow throughout the network while addressing goals such as emissions reduction and pedestrian safety. Additionally, the RL framework can streamline initial traffic light scheduling for new intersections, minimizing the need for labor-intensive adjustments and allowing engineers to focus on strategic planning.

Beyond operational efficiencies, this integrated approach promotes safer, more sustainable, and equitable transportation systems. By addressing both immediate traffic management issues and long-term urban development goals, the framework contributes to cleaner air, safer streets, and vibrant public spaces, ultimately improving the well-being of urban communities. Thus, this transformative framework not only modernizes traffic systems, but also enriches urban life across multiple dimensions.

1.6. Study setup

This study conducted four core experiments to explore the potential of reinforcement learning and graph theory to optimize urban traffic flow in Almelo, Netherlands, addressing key research questions regarding KPI evaluation, intersection performance, and policy-driven optimization. **Chapter 2** provides the overall framework used for this research, detailing how all research questions relate to each other and interact.

First, graph theory was applied to identify traffic bottlenecks and key intersections for improvement. While effective at pinpointing static bottlenecks, this method proved insufficient for capturing dynamic traffic behaviors such as weaving. This aligns with the sub-question on how intersections can be identified as bottlenecks using graph-theoretical methods under specific traffic conditions or demand patterns. The chapters discussing these aspects are **3.2, 4.2, and 5.2**.

Second, a local RL agent, trained using Deep Q-Networks (DQNs), demonstrated that stable and efficient learning could be achieved with properly tuned reward functions and discount factors; however, scalability remained a challenge. This experiment addresses the sub-question related to evaluating performance indicators at individual intersections and the effectiveness of RL in real-time traffic management. For details on the local agent, refer to chapters **3.4, 4.3.2, and 5.3.1**.

Third, the introduction of a global RL agent to coordinate local agents significantly improved network-wide performance. It balanced conflicting KPIs, such as car flow, emissions, and safety, aligning with policy goals for sustainable urban mobility and efficient traffic flow. This speaks directly to the main research question about applying RL algorithms to optimize traffic light control in a way that supports policy goals. More on the structure and implementation of the global agent can be found in chapters **3.4, 4.3.3, and 5.3.2**.

Finally, various policy-driven scenarios, including car-focused and balanced setups, were tested to assess the adaptability and real-time responsiveness of RL. Results demonstrated global coordination's capacity to maintain overall network performance while supporting long-term municipal goals. This addresses both short-term and long-term sub-questions related to the effectiveness of RL-based traffic light controllers in fluctuating traffic conditions and optimizing traffic light scheduling in support of long-term policy goals. Relevant chapters include **3.1, 4.1, 5.3.2, and 5.4**.

By designing these experiments, the study systematically explores how KPIs can be evaluated, how individual intersection performance can be aggregated, and how RL algorithms can support both immediate and long-term traffic management objectives. These experiments provide a holistic approach to understanding and optimizing urban traffic flow while aligning with broader societal goals. **Chapter 6** presents the final remarks and suggestions for future research to further improve the current findings.

2

Translating KPIs into traffic light control: a bidirectional framework

Efficient traffic management is crucial for urban areas. This chapter presents a bidirectional framework to translate Key Performance Indicators into optimized traffic light control using reinforcement learning and graph theory. Bidirectionally refers here to the fact that one influences the other and back so it is a two way street. This chapter will show the loop that exists between scoring and the RL model which gets updated by the scores from the dashboard. The RL agent can than optimise this score. When the weights are changed the scoring changes and the final result of the agents can look different than it would under different policy goals.

Section 2.1 introduces the conceptual framework, detailing the use of detectors to calculate KPIs and integrate computational codes to evaluate and optimize traffic performance. Graph theory helps assign weighted importance to different road segments. Section 2.2 covers the design and implementation of the dashboard, scoring systems, and RL algorithms, explaining how they work together to balance municipal priorities. Section 2.3 outlines the experimental design using PTV Vissim for traffic simulation, comparing traditional schedules to RL-optimized ones at both intersection and network levels.

2.1. Introduction conceptual framework

To effectively measure Key Performance Indicators in transportation networks, it is essential to consider different spatial levels. Travel-time KPIs, for example, require a network-wide analysis to provide meaningful insights into overall network performance. In contrast, KPIs such as waiting time at intersections are more relevant at the local level. The spatial reach of pedestrians and cyclists is much smaller than that of cars, which further requires scoring performance at different spatial levels.

To address this, the analysis is split into two layers: network level and intersection level. Each KPI is appropriately assessed within its required spatial context. To combine these different levels into a single comprehensive score for reinforcement learning input, weights will be assigned to each intersection and these scores will be aggregated. How these intersections are summed will be explained in a much more detail later. For research, both the network and the intersection levels are given equal importance (50/50) to ensure balance.

The conceptual framework, illustrated in Figure 2.1, delineates the essential components of this research. The simulations will feature detectors to calculate various Key Performance Indicators (KPIs). The simulation environment, represented as the digital twin, addresses the link loading aspect within the framework (see Figure 2.1). The KPIs utilized are part of the first sub-question, and these KPIs are directly derived from the simulation itself.

These different KPIs will be employed across three separate modules: Code I (Scoring of the Intersection), Code II (Scoring of the Network), and Code III (Intersection Weighting within the Network), as illustrated in Figure 2.1. Code III will serve as the integrative component between Code I and Code II.

This is the section where graph theory is introduced to determine the relative importance of intersections or road segments. Graph theory will be applied to assign weights to the significance of various intersections or road segments.

For instance, many routes utilize tunnels or bridges; if these routes are blocked, many commuters will experience delays. This example underscores that not all road segments or intersections should be weighted equally. Although graph theory is well-studied [8], [9], [10], its application for this purpose represents a novel contribution in this study. The scores will be computed using weights set by the municipalities themselves to align with their short-term policies. Thus, this tool can aid municipalities in implementing their policy objectives effectively.

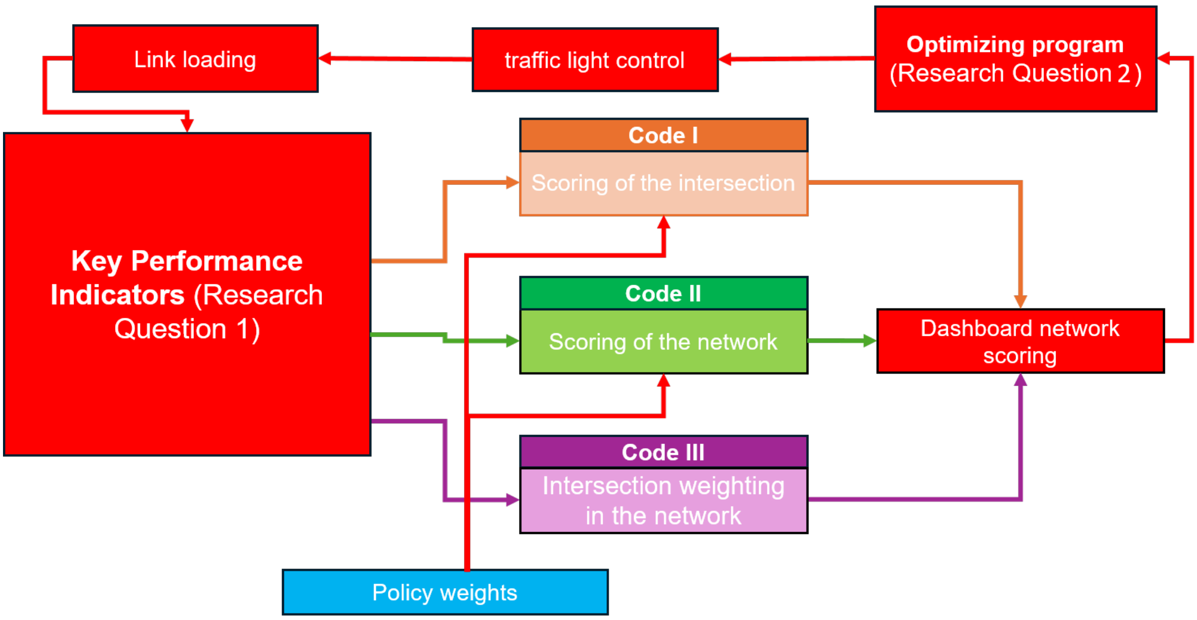


Figure 2.1: Introduction into conceptual framework

The KPI scores serve two primary purposes: dashboard network scoring and input for the optimizing program (see Figure 2.1). The KPIs are either obtained directly from PTV Vissim or sourced from literature (see Chapter 3.4). While the calculation of KPIs follows established methods, their combination and comparison represent a novel approach in this study.

Three different scoring methodologies will be proposed. The primary methodology will be explained in detail, while the other two will only be briefly outlined. The other two are alternative option of scoring the KPIs but will not be used for the research so the idea behind them will be explained. This program incorporates a reinforcement learning model that adjusts traffic light control parameters to enhance network-wide performance. The reinforcement learning algorithm will optimize traffic light control to achieve the highest total combined score. The traffic light system will employ a free control structure with constraints. Policy changes will directly affect the scores, thus impacting the optimization process and prompting the software to identify a new optimal solution.

As stated in the scope of this research, the focus is solely on the morning peak, which limits the total training time per intersection.

For this research, a digital twin will be used as a virtual representation of the real world traffic environment, this way the link loading can be done more accuratly. A well-designed digital twin can closely mirror real-world conditions. Extensive research has shown that digital twins can serve as accurate and reliable simulations of physical systems (for more proof, see the Appendix A). Given the high fidelity of the simulation, reinforcement learning models trained within this environment can be transferred to real-world deployments with minimal adjustments or performance issues.

In conclusion, this research introduces a novel bidirectional framework that leverages reinforcement learning and scoring of key performance indicators to optimize traffic light control based on these KPI score. While the calculation of individual KPIs follows established methods, the innovative aspect lies in their combination and comparison to enhance traffic management. This study uniquely integrates graph theory to assign weighted significance to different road segments and intersections, a methodology that departs from traditional approaches. Furthermore, the use of a digital twin for accurate link loading and simulation further bolsters the robustness of reinforcement learning models. By aligning KPI scores with municipal policies, this framework offers a dynamic tool for policy implementation and traffic optimization. Thus, this research stands out by combining established techniques with novel applications to address complex urban traffic management challenges.

This research presents a novel, two-level reinforcement learning framework designed to optimize traffic light control by incorporating broader societal goals, such as emissions and noise, alongside traditional metrics like waiting times and vehicle throughput. Existing hierarchical RL frameworks typically focus solely on minimizing delays and waiting times at individual intersections or throughout the network [11] [12], [13]), often neglecting dynamic interactions and broader impacts throughout the network on societal goals as a whole. The proposed framework addresses this gap by introducing a global observer and local node agent. This dual-level approach ensures that immediate operational efficiency is balanced with longer-term, network-wide policy objectives.

The local agent updates traffic signals in real-time at one-second intervals, while the global observer is updated less frequently to maintain a holistic view. By integrating periodic evaluations and aligning with the final score derived from multiple Key Performance Indicators, this research aims to provide an optimal traffic management solution that serves both localized and network-wide societal needs.

This innovative methodology extends the typical scope of RL frameworks, demonstrating a significant advancement in urban traffic management by balancing efficiency with broader societal objectives.

2.2. Model setup

- **Dashboard Design:** A dashboard will visualize network performance and KPI scores. Features such as adjustable sliders will allow users to weigh KPIs, aligning results with specific municipal priorities.
- **Simulation Framework:** A simulation model will be created to mirror urban traffic dynamics. Firstly, VisVap[1] will be used to create a reference traffic light schedule. Reinforcement learning will be used to optimize traffic light control, running iterative experiments to maximize network-wide KPI scores.
- **Scoring System Development:** Three tools will be developed that feed into the dashboard:
  - **Code I:** Evaluates KPIs for individual intersections.
  - **Code II:** Assesses the performance of the network.
  - **Code III:** Weighs the importance of intersections based on their role in the network. Also the importance of a stretch of road is calculated. The results of these tools will be combined to produce a comprehensive performance score.
- **Reinforcement learning for traffic signals** The software will be designed to use input from a scoring system to effectively guide the deep Q network (DQN) in optimizing traffic signal stages. Through reinforcement learning, the software will continuously learn and adapt. A local agent will focus on intersection throughput and delays. This local agent will be bounded by safety guidelines to prevent it from making dangerous/illegal decisions. Each intersection will have a separate local agent that operates independently of the others. A global agent will instruct/influence the local agents to consider overarching goals such as emissions, which impact a higher level. Because the global agent gets as input the final score (see dashboard Figure 2.1), therefore the global agents gets more control of the overall performance given the weights for each intersection/road

[1]VisVAP enables the use of object-oriented programming. The logic of the traffic control algorithm is implemented using flowcharts. An example can be found in Appendix B

segment. This could mean that the local agent is forced to adapt a suboptimal structure to satisfy the global agent.

- **Case Studies and Validation:** The methodology will be applied to Almelo (a relatively small city in the Netherlands (75k inhabitants)), testing policy scenarios and their optimal traffic light schedule (comparing man-made signals vs RL signals).

2.3. Experimental setup and evaluation

The simulation environment is based on a network model comprising several connected intersections, developed by Sweco. This model reflects a realistic urban traffic layout and will serve as the foundation for evaluating various traffic signal control strategies.

PTV Vissim will be used as the primary microscopic traffic simulation tool to simulate vehicle movements and intersection-level dynamics. Although Vissim may not be the best tool available, it is currently widely used by many municipalities, including Sweco's already calibrated model for Almelo, which will serve as the case study. Using Vissim ensures that the improvements we develop are immediately applicable and beneficial to communities that already use this tool. To establish a baseline, a static traffic signal schedule will be generated using VisVap, which provides a rule-based, demand-responsive control logic (for further information on this see Appendix B).

In contrast, a deep Q-Network-based reinforcement learning algorithm (DQN) will be developed to dynamically optimize traffic signal timings. The reason for choosing a DQN-based approach lies in its ability to adaptively refine traffic management strategies by learning real-time data, thus achieving superior results compared to static models. This neural network will learn optimal signal control policies through interaction with the simulated environment, guided by a reward function based on KPIs such as vehicle delay, queue length, emissions, and throughput, to name a few.

To assess the impact of the proposed approach, performance will be evaluated at two levels:

- **Intersection-Level Evaluation:** A single intersection will be isolated and optimized using both VisVap and the DQN-based method. The resulting performance metrics will be compared to demonstrate the improvements achievable through reinforcement learning at the intersection level.
- **Network-Level Evaluation:** The full connected network will be tested with three traffic scenarios corresponding to three totally different types of policies. These scenarios will differ by adjusting the weights assigned to different KPIs to simulate various municipal priorities (e.g., prioritizing emissions reduction versus minimizing travel time).

A custom dashboard will be used to visualize and compare the performance of each control method in different scenarios. This tool will allow users to dynamically adjust KPI weights and observe corresponding changes in overall network efficiency. The improvements of the RL-based control over traditional VisVap schedules will be highlighted using the dashboard's comparative analytics features.

To ensure robustness, additional experiments will be conducted to evaluate the sensitivity of the RL model to changes in traffic demand and signal timing constraints. The computational performance and training convergence of the RL model will also be documented.

3 Literature review

This review of the literature examines key research on traffic performance indicators, network bottleneck identification, and intelligent traffic control systems. It highlights how data-driven methods and graph-based approaches are used to evaluate and improve urban mobility. The aim is to provide a foundation for developing efficient and adaptive traffic management solutions.

3.1. Identification of relevant KPIs for policy goals

The European Union has conducted extensive research into the development of performance frameworks for traffic management and Intelligent Transport Systems (ITS). In a report published by POLIS Network [14], four main pillars are identified as the foundation for assessing traffic system performance:

1. **Traffic Efficiency**
2. **Safety**
3. **Social Inclusion & Land Use**
4. **Pollution Reduction**

These pillars provide a comprehensive framework for evaluating the impacts of traffic systems on urban environments. In support of this framework, the University of London offers additional empirical analysis that emphasizes the importance of reliability [15] as a key performance indicator in traffic systems [14].

A widely adopted KPI for evaluating congestion is based on travel time delays and the monetary value of time, which helps to quantify network performance and identify areas for improvement [16]. European cities are increasingly advocating for more robust and holistic performance frameworks for their transportation networks.

In the Netherlands, the share of cyclists and pedestrians is substantial and continues to increase. Urban planning efforts are increasingly prioritizing these active mobility. A notable example is the *STO(M)P-principle*, which outlines a hierarchy in transport planning:

> *Cyclists and pedestrians are receiving increasing attention and space. There is a growing focus on how to divide the scarce space in cities and on promoting active mobility. The STO(M)P principle plays a key role in this. It prioritizes pedestrians first (Stappen – walking), then cyclists (Trappen – cycling), followed by public transport and Mobility-as-a-Service (MaaS), and lastly private vehicles.* [17]

Dividing performance indicators into these four categories is essential to identify where the most significant improvements can be made. As E. Dumbaugh notes, city streets should not simply be “forgiving”—they must be *designed with safety in mind* from the design.

Although vehicle emission pollution tends to receive the most attention, the impact of *noise pollution* has also been extensively studied. Chronic exposure to traffic noise has been shown to pose serious health risks [18][19][20]. Today, awareness of these effects is growing, and many public health authorities now recognize noise pollution as a serious concern. Documented health consequences include:

> *Sleep disorders, learning impairment, cardiovascular diseases, metabolic syndrome, hypertension, ischemic heart disease, increased risk of diabetes, obesity, and general annoyance* [21].

3.1.1. Network related key performance indicators

The following paper written by I. Kaparias wants to assess other factors, these factors range from: ease of access between certain representative origin-destination pairs, safety impacts, emissions originating from traffic, social inclusion and land use, to name a few [14]. Other research names factors such as vehicle and pedestrian delay related measures, average crash frequency, and pedestrian delay. Recent advances in traffic signal performance evaluate, use of crash surrogates, and exceedance statistics to estimate road safety. These KPIs show that the traffic system is more complex than only ”hard” performance indicators can capture. Hard performance indicators, such as travel time, congestion levels, and throughput, provide quantifiable measures of efficiency. However, they often fail to account for softer aspects such as equity, environmental impact, or neighborhood livability, which require a more nuanced multidimensional evaluation.

3.1.2. Intersection related key performance indicators

Analysis of key intersection performance indicators has received significant attention in traffic management studies. A critical KPI (I_{IS-U}) evaluates the number of critical situations (e.g., delays or cycle times) at intersections and their impact on traffic flow. This metric accounts for the weight of each link in the network and is calculated using the number of critical situations detected (CS_l) normalized by the daily traffic volume (DTV_l) [14]. However, consistency issues arise from the varying definitions of ”critical situations” and differing thresholds between cities. Therefore a standardised approach will be key to compare these different designs.

This highlights the need to improve signal settings to minimize congestion. Frequent adjustments to signal timings are often required during peak periods, with the frequency of changes depending on the demand characteristics [22]. These approaches distinguish various traffic states, enabling more accurate congestion assessments. Expanded models address practical challenges such as limited data resolution or combined detector inputs, demonstrating acceptable accuracy compared to ground truth data.[23].

Intersections with traffic lights are implemented to enhance safety and throughput by preventing crashes through the separation of conflicting traffic flows. Hameed A. demonstrated in his virtual reality experiment that red lights were ignored 3.7 percent of the time. Of these violations, 60 percent occurred just after the red light appeared, indicating that people often trust the built-in safety margins and the buffer time incorporated when signal groups change [24]. In the municipality of Enschede, the concept of the “first waiting pedestrian/cyclist” is applied to improve both safety and equity at intersections [25]. Increasing the number of green times for these users has been shown to reduce the number of red light violations, highlighting that minimizing cycle time is not just a convenience, but a necessity [26].

However, the effectiveness of a bicycle-friendly signal regime is highly dependent on the volume of cyclists and the pressure exerted by other traffic participants at the intersection. According to the Fietsberaad study, when this pressure becomes too high, waiting times for other road users can increase significantly. It remains unclear at what point the disadvantages in waiting times for non-cyclists outweigh the benefits provided to cyclists. This underlines that throughput for all users (cyclists, pedestrians, and motorized traffic) is a crucial key performance indicator in intersection design and traffic light management.

The last user groups that require consideration are emergency vehicles and public transport. For emergency vehicles, reliable and fast passage through intersections is essential to enable firefighters, medical personnel, and other first responders to perform their duties effectively [27]. In contrast, public transport prioritizes service reliability over speed. For example, if a bus arrives early, waiting slightly longer at a red light can help maintain a consistent schedule, which can actually benefit overall reliability. A study carried out in the Netherlands found that improvements in bus scheduling can be achieved without additional cost, although this can result in increased delays for other road users [28].

3.1.3. Different views on key performance indicators

Municipalities play a key role in designing and implementing transport policies that are influenced by a variety of political ideologies and public value frameworks. The choice and weighting of KPIs such as safety, congestion, emissions, and economic efficiency often reflect the underlying values of the governing political party and whether policy design is guided by an individualistic or a collective perspective.

For instance, left-leaning political parties, which prioritize environmental and social justice, are more likely to focus on KPIs such as emissions reduction, public health, and transport accessibility. This focus can translate into municipal policies like:

- Low-emission zones to improve air quality [3].
- Expanded cycling infrastructure and public transit subsidies to encourage sustainable transport modes [29].
- Traffic calming measures to enhance pedestrian safety in residential areas. [30]

In contrast, liberal or economically focused parties tend to emphasize economic efficiency, congestion mitigation, and cost-benefit ratios. Municipal policies under such leadership may prioritize:

- Smart traffic management systems to optimize flow and reduce congestion [31].
- Public-private partnerships for infrastructure investment [32].
- Road pricing or dynamic tolling to internalize the economic costs of road use [33].

Furthermore, research by Rienstra et al. (1999) highlights a shift in KPI prioritization when individuals are prompted to consider societal rather than personal impacts. From a social perspective, safety emerges as the top priority, followed by congestion and then emissions. Municipalities that adopt a socially-oriented policymaking framework can therefore design initiatives that align with this hierarchy. Policies might include:

- Vision zero strategies to eliminate traffic fatalities and serious injuries [34].
- Integrated public transport and mobility hubs to reduce urban congestion [35][36].
- Urban greening and electric vehicle incentives to address emissions [37].

By explicitly linking transport policies to prioritized KPIs, municipalities can ensure that their initiatives are not only ideologically coherent, but also responsive to the public's values, whether viewed through an individual or collective lens. This KPI-driven approach to policy also enables more transparent performance tracking and accountability.

3.2. Uses of graph theory to find bottlenecks in the network

Traditionally, the fields of network design and traffic management have operated separately. However, researchers are increasingly recognizing that closer collaboration between these domains can lead to more effective solutions [38]. For example, G. Sarlas demonstrates that dividing a network into distinct zones can provide valuable insight into how different parts of the system interact. This zonal approach enables the identification of congestion hotspots by analyzing the network topology using betweenness centrality (BC, see 3.1) [39].

This method leverages graph theory in combination with traffic measurement data to deliver a more integrated assessment of traffic systems. One of the key contributions of this approach is the integration of multiple key performance indicators, which are often considered in isolation. Graph theory helps identify the most critical sections of the network, allowing for more accurate, weighted KPIs to be incorporated into the reinforcement learning framework.

For example, consider two stretches of road, one significantly more important to the network than the other. The reinforcement learning algorithm would assign greater weight to the more critical segment, leading to its performance improvement, even if it means a slight degradation in the less important segment. This prioritization reflects a more strategic optimization of the overall network performance.

The indicators that will be used will range from once focused on the entire network to others more zoomed in and focused on how a single intersection performs. [40] Mohammad Reza adds that using graph theory allows the computer to find the theoretically most critical intersections. [41] The centrality of a given vertex in the graph can help to find the weak points in the network. This is done by a measure based on the number of paths that pass through it, this is called betweenness in the literature [39] [10].

$$b_i = \sum_{s \neq t} \frac{n^i_{s,t}}{n_{s,t}} \tag{3.1}$$

Where $n^i_{s,t}$ are all the shortest paths through link i going from the OD pair s to t and $n_{s,t}$ are all the shortest paths from s to t (where s and t are all the origin and destination points). Therefore, a bridge, for example, will have many shortest paths traversing this link. This shortest path does not take into account alternative shortest paths that could be a valid alternative and just a fraction longer. C. Mylonas adds that this can be multiplied by the volume of traffic to obtain a better value called the centrality of the weighted betweenness of the volume of traffic [8]. When shortest paths are relatively similar, they can be seen as a valid alternative and reduce the importance of that one link. The Wardrop assumption that travelers choose the fastest available route is an example of an idealization [42]. To see which node is the most important clustering is a very strong tool in networks to find the nodes that connect to many people. In graphs that focus on relation between people, these are the people that connect groups of people together. In the COVID-19 period this was called superspreaders [43], in other words, the people who bridge friends' groups. In the traffic network, these are the intersections that connect the neighbourhoods to the larger network [9] [10]. As can be seen in Figure 3.1 the 2 nodes connecting both groups are much more important for the network than the other nodes.

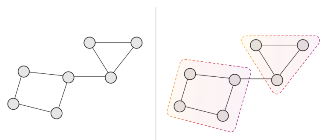


Figure 3.1: Graph clustering visualization [44]

The Python library from netwrokx has already extensive tools to find these most important nodes [45]. This library allows for efficient finding the value of connectivity for all nodes in the network (values range between 0 and 1). A low value represents an important node, the higher the value, the more connected it is, and therefore has more alternative paths in case of disruptions.

3.3. Traffic light control

Traffic lights play a central role in managing urban traffic flows. They balance conflicting traffic movements across directions, prevent gridlock, and ensure intersection safety. Interestingly, the concept of the traffic light dates back to early systems used to guide fishermen back to shore [46]. Since then, traffic light systems have evolved through four key stages.

The first stage, as described in Smart Cities and the Importance of Smart Traffic Lights [47], is the fixed signal controller. This system operates on a pre-defined cycle with fixed durations for green and red phases, regardless of real-time traffic conditions. As a result, it can be inefficient, granting green time to directions where no vehicles are waiting.

The second stage introduces adaptive timing using loop detectors embedded in the road. These detectors allow the traffic signal to adjust green phase durations based on actual demand. Moreover, signal groups can be skipped entirely if no vehicles are present, significantly improving efficiency over the fixed-time model.

The third stage is characterized by queue-length-based coordination. At this level, traffic signals respond to the length of vehicle queues at intersections. Furthermore, there is communication between adjacent intersections, enabling coordinated control strategies along traffic corridors to improve overall performance.

Finally, the fourth stage, which is the focus of this research, involves network-level coordination. In this advanced system, traffic signals adapt based on traffic demands across multiple intersections simultaneously. The goal is to optimize performance across the entire network rather than at isolated nodes. This level of control also incorporates broader KPIs, including emissions, safety, and equity. It aims to create a more holistic and sustainable urban traffic management system.

3.4. RL to optimize traffic lights

Reinforcement learning has emerged as a promising solution for adaptive traffic signal control in urban environments, addressing limitations of fixed-time systems. Recent surveys highlight increasing interest in RL from both transportation and computer science fields, categorizing various approaches and simulation tools like SUMO for performance evaluation [48].

Adaptive Traffic Signal Control (ATSC) systems powered by RL are crucial for managing congestion in smart cities. Multi-agent architectures and several RL algorithms have shown effectiveness in improving traffic flow [49]. Hierarchical Reinforcement Learning (HRL), in particular, offers adaptability by breaking down complex tasks, outperforming traditional methods such as the Webster-based fixed-time controllers [12].

Multi-agent RL systems with hierarchical structures and predictive capabilities using long and short term memory (LSTM) networks have proven effective in coordinating multiple intersections, improving traffic network efficiency [50]. A novel two-level HRL approach further enhances performance by optimizing green phases and managing low-traffic periods with flashing modes, significantly reducing delays and fuel consumption [51].

Reinforcement learning, especially in its hierarchical and multi-agent forms, shows significant promise for intelligent traffic management. However, current models still face important limitations. Most RL algorithms are primarily designed to optimize vehicle flow, often neglecting broader performance metrics such as emissions, noise pollution, and equity. Moreover, safety is a critical concern: RL algorithms must operate within predefined safe bounds to prevent dangerous situations from arising.

When additional factors like environmental impact or social fairness are considered, they are usually assessed in isolation rather than within an integrated, multi-objective framework. This siloed approach restricts the real-world applicability and long-term sustainability of many existing RL-based traffic solutions.

3.5. Summery and implementation for this research

This review of the literature has provided a detailed exploration of current research and methodologies on traffic performance indicators, network bottleneck identification, and intelligent traffic control systems. The insights gathered from this comprehensive review will be pivotal in the direction of the following chapters of the thesis.

Initially, the identification of relevant KPIs for policy goals will inform the criteria used to evaluate and optimize the proposed traffic management solutions. The diverse range of KPIs, spanning efficiency, safety, social inclusion, and pollution reduction, will ensure a holistic consideration of the impacts of the urban traffic system.

The findings on network-related KPIs and intersection-specific indicators lay the foundation for developing sophisticated traffic assessment models. These models will be crucial in pinpointing specific areas within urban environments that need improvements, thus facilitating targeted interventions. The multifaceted nature of these KPIs underscores the need for a comprehensive approach that balances quantifiable measures such as travel time and congestion with qualitative aspects such as equity and environmental sustainability.

The application of graph theory to identify network bottlenecks presents a robust method to scrutinize urban traffic systems. The zonal approaches and centrality metrics discussed will be instrumental in locating critical nodes and links that contribute to congestion. These insights will be used to design traffic management algorithms that prioritize strategic network segments for optimization. This research will expand the current mathematical formulation that exists surrounding these topics.

Insights into traffic light control systems' evolution illuminate the importance of adaptive and network-level strategies. The existing stages, from fixed signals to advanced network coordination, highlight the inefficiencies of traditional models and the potential of innovative approaches. This understanding will drive the development of a novel traffic light management system that integrates broader KPIs and adapts dynamically to real-time urban traffic demands.

Furthermore, RL methods offer promising adaptive traffic signal control solutions. The literature points to the efficacy of hierarchical and multiagent RL systems in enhancing traffic flow and network efficiency. These advanced algorithms will be used to create a scalable and responsive traffic management framework. This framework will go beyond vehicle flow, incorporating environmental, social, and safety metrics into optimization processes, offering a more sustainable and equitable urban mobility solution.

In summary, the insights from this literature review will be applied systematically throughout the remainder of the thesis. By integrating data-driven KPIs, sophisticated graph-theory methods, adaptive traffic control systems, and RL-based optimization, the thesis will propose a series of innovative solutions aimed at significantly improving urban traffic management.

4 Model setup

This chapter outlines the essential framework and methodologies for evaluating and optimizing traffic network performance. Building upon the comprehensive review of the literature in Chapter 3, it dives into the specific themes, algorithms, and tools that underpin the model (outlined in Chapter 2). The chapter is structured to guide the reader through the conceptual foundations of the framework, the technical implementation details, and the practical applications of RL for traffic management.

Section 4.1, Introduction Framework: This section lays out the overall framework for scoring and optimizing the transportation network. Identifies critical themes, including car traffic, bicycle traffic, public transport, pedestrian traffic, noise, road safety, air quality, and equity, and explains how they are incorporated into the model using adjustable sliders. This part is particularly relevant for policymakers, as it includes a visual representation of the iterative feedback loop and demonstrates how policy adjustments impact the scoring mechanism. It also explains the rationale for maintaining constant parameters, such as the OD-matrix and mode split, to ensure consistency.

Furthermore, this section delves into how metrics are calculated in Subsection 4.1.3, explaining the procedures for converting values into scores ranging from 0 to 10. This subsection also outlines the setup of the three different scenarios and how they relate to each other. In the final part of the section (Section 4.1.4), the proposed dashboard functions are visualized to provide users with a clearer idea of how the tool works and how they would interact with it.

Section 4.2, Graph Theory Learning: This section focuses on how graph theory is applied to identify and prioritize network bottlenecks and critical nodes. It involves understanding the strategic importance of various intersections and road segments. This part of the chapter is especially relevant for traffic controllers, who need to be aware of potential bottlenecks and which parts of the network require extra attention.

Section 4.3.3, Reinforcement Learning: This section is divided into two subsections. It highlights the mathematical formulation of the RL approach and includes a textbox to assist readers less familiar with the topic in understanding how the artificial intelligence agents are designed and behave:

- **Local agent:** Details the environment and mathematical formulation for optimizing traffic flow at intersection level, including action space, reward mechanisms, and state transitions.
- **Global agent:** Discusses the broader environment and design considerations to optimize the overall performance of the network using grid search and parameter optimization.

By the end of this chapter, a clear understanding is established of how the identified themes and KPIs from the literature review are translated into an operational model. This model leverages reinforcement learning to dynamically manage and optimize urban traffic systems.

4.1. Introduction framework

Based on Chapter 3, eight key themes were identified as essential to evaluate network performance: traffic by car, bicycle, pedestrian traffic, public transport, road safety, air quality, noise, and equity. The first seven themes are considered the most directly relevant for scoring a transportation network. The eighth theme, equity, is represented differently in the framework because it interacts with all user groups as well as with other themes. For example, noise exposure, which may disproportionately affect lower-income neighbourhoods due to poorer housing insulation.

Although equity is not yet fully implemented in this research due to its inherent complexity, it remains a crucial aspect and is acknowledged as an important factor for future integration. In the current framework, equity is assigned its own slider, allowing it to influence specific KPIs within the other thematic categories. Each of the eight themes is equipped with an adjustable slider, enabling policymakers to define the relative importance of each theme based on their goals. Sliders were deliberately chosen for their simplicity and user-friendliness. They provide an intuitive way to experiment with policy trade-offs without requiring technical expertise. Moreover, this design lays the foundation for more advanced integration in the future.

This weighting process could be automated through tools such as questionnaires or large language models. For instance, a user might simply input a command like: "I want a healthier city with more cyclists and pedestrians," and the system would automatically adjust the theme weights to align with that vision. This would make the tool even more intuitive and accessible for municipal users.

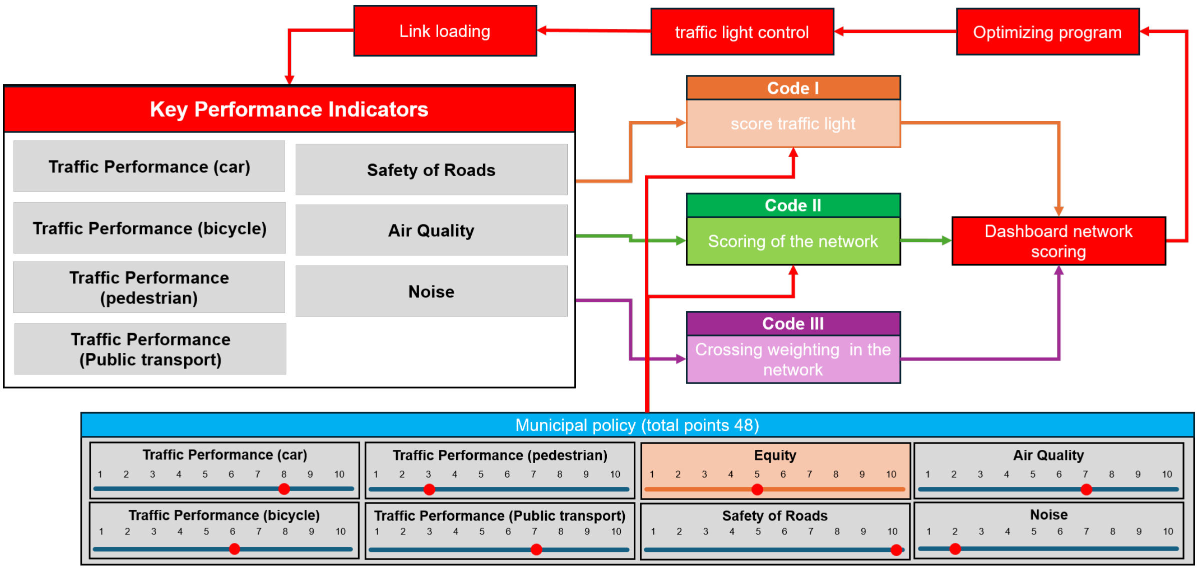


Figure 4.1: Framework overview

The framework (Figure 4.1) illustrates an iterative feedback loop in which policy adjustments influence the scoring process. Based on updated sliders, the optimum is recalculated and a new traffic light control scheme is generated. Then, this scheme is presented alongside its impact on various road users and different parts of the network. However, for the purposes of this research, certain parameters, such as the OD-matrix and mode split, remain constant, as previously mentioned. The latter helps to improve consistency between different runs. The goal is to show that a holistic approach achieves better results than looking at each intersection individually. The traffic lights are not only optimized for throughput or traffic flow, but they are more focused on giving the best optimal solution for the entire network. Reinforcement learning should help achieve the goals in the best way possible, as some of the parameters are very complex. A deep neural network should be able to make sense of this complexity.

4.1.1. Future Improvements to the dashboard

While the current dashboard allows users to set the relative importance of policy themes through adjustable sliders, there are several areas that could be enhanced in future iterations.

First, the dashboard does not yet support customization based on user groups (e.g., commuters, children, elderly, or people with limited mobility). Although the equity theme is included in the framework, it currently operates at a high level and does not capture the nuanced needs of different population segments. Future research could expand on this by incorporating user-specific KPIs, enabling a more inclusive and socially responsive optimization process.

Second, while the implemented graph theory methods help prioritize intersections and road segments based on their strategic importance within the network, the dashboard currently does not allow for manual selection of specific roads or intersections by the user. Adding this feature would give municipalities and planners more direct control, especially in cases where local knowledge or context-specific concerns (e.g., construction zones, schools, or hospitals) need to override algorithmic prioritization.

Incorporating these enhancements would significantly increase the dashboard's flexibility and make it a more powerful decision-support tool for cities aiming to tailor mobility policies to both spatial and social realities.

4.1.2. Combined framework

When integrating the different KPIs into the overall framework, the complete problem can be visualized (see Figure 4.2). This figure provides a comprehensive picture of the problem. The pink boxes highlight the two mean research questions. The box on the right represents the dashboarding system (number 1), designed to make the transport system easier to understand. It will show a single score for the network, this score will give a meaningful value to see if the network improves or worsens. Meanwhile, the box on the left illustrates the reinforcement learning programs (number 2), which will optimize the score based on the assigned weights. This will be the structure of the traffic lights that leads to the scoring of the KPIs. This loop shows that one cannot function without the other. Therefore, first the scoring question has to be solved before reinforcement learning can start improving the overall score. The blue box shows parameters that are left outside the scope of this research.

The KPIs are represented using three different colours:

- **Orange:** Indicates the KPIs related to the performance of a single intersection.
- **Green:** Indicates the KPIs related to the overall performance of the traffic network.
- **Pink:** Represents the importance measure of each intersection or street. Graph theory is used here to determine which intersections are more important than others.

As mentioned in Chapter 2 (Methodology), Code III will be used to combine Score 1 and Score 2. The score of each intersection will be multiplied by its corresponding importance factor. Similarly, some network KPIs, such as speed, will be multiplied by the street importance factor to give more weight to streets that are deemed more important. These final network and intersection KPIs are then summed to produce a final score. For this research, all KPIs within each theme are considered equally important, although this weighting could be adjusted with minimal effort.

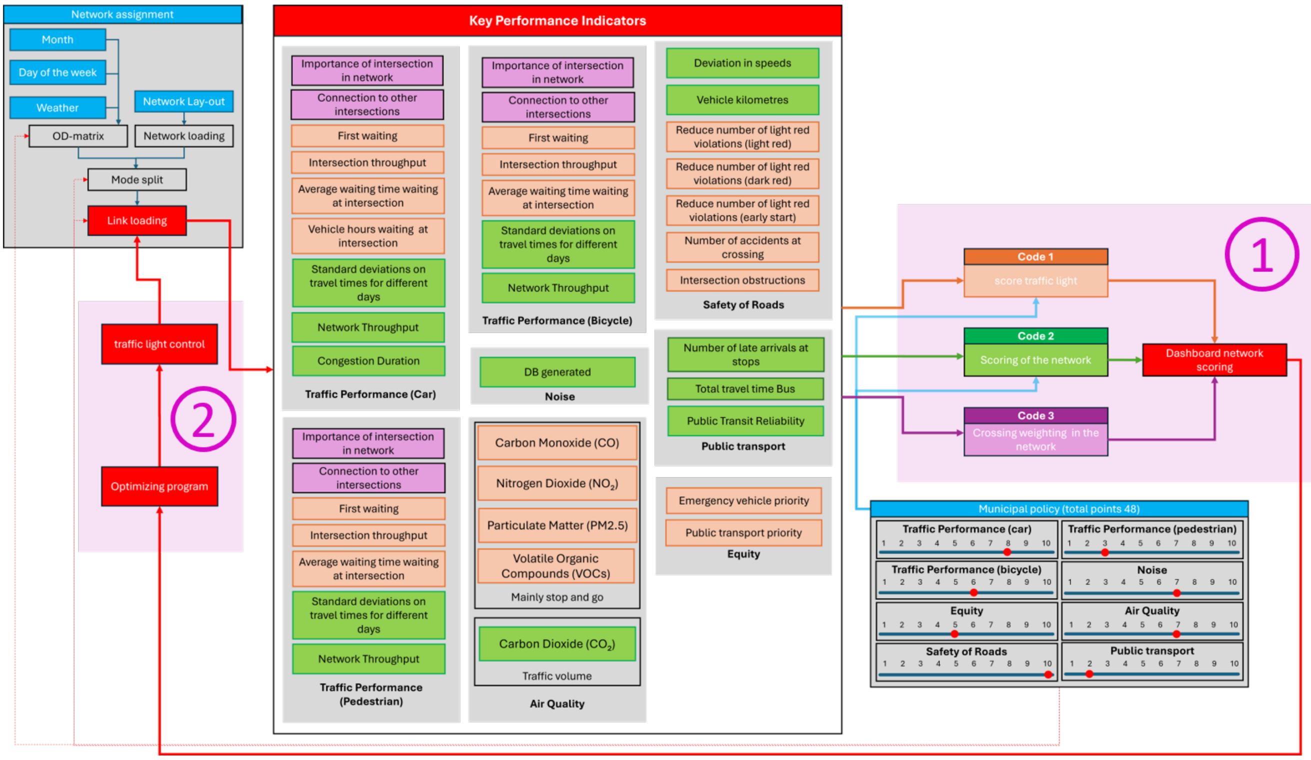


Figure 4.2: Conceptual framework, number one indicates the dashboarding research question and number two shows the RL based on the results from the dashboard

4.1.3. KPI calculation

To enhance the results and store data for different days, PTV VISSIM is linked to a Microsoft Access database, which organizes all results into separate tables for each run. The runs used will be explained at the end of this chapter. Python is then used to combine these various results, generating outputs for each intersection. Network-wide performance metrics are provided at three different levels: the entire network, each neighborhood, and some metrics that are even specific to individual links, such as emissions and speeds.

For emissions, it would be valuable to incorporate the distance to people's home/work as a weighting factor; this approach could also be applied to noise levels. Although this is beyond the scope of the current research, it should illustrates that integrating these factors could significantly enhance the tool's effectiveness.

4.1.3.1. Calculation

There are three different methods devised to convert the values into scores that can range from 0 to 10. The three methods are explained in more detail in Appendix C, method 3 will be explained in more detail below as it was also applied for this study, the other 2 are primarily conceptually developed.

Table 4.1: Example converting KPIs value into scores

Travel Time (TT)	Score
11.00	5.50
5.00	7.50
12.00	5.17
8.00	6.50
20.00	2.50
Total	**5.43**

The original scenario is scored for all time intervals together (time interval five 5 min for this research). The lowest value is equated to 2.5 and the maximum value is equated to 7.5. The remaining values are assigned a value through linear interpolation between them. Below, a simple example is worked out:

The new scenario is then scored using the same scale as the old scenario, which ranges from 0 to 10. Because both scenarios utilize the same scale, comparing them becomes much simpler and more insightful. It's important to note that if the spread (range of scores) is very small, even slight improvements can result in significant differences in scoring values. The overall score is calculated by taking the average of all values.

Although more elaborate methods can be adopted to give more weight to busy periods, for simplicity, the average is used. As seen from the table below, the overall scores have improved; however, the improvement is not as significant as one might expect when looking solely at the absolute values.

Table 4.2: Example converting KPIs value into scores new and old scenario

Time	TT (OLD)	Score		TT (NEW)	Score
08:00:00 - 08:04:59	11	**5.50**		10	**5.83**
08:05:00 - 08:09:59	5	**7.50**		3	**8.17**
08:10:00 - 08:14:59	12	**5.17**		13	**4.83**
08:15:00 - 08:19:59	8	**6.50**		6	**7.17**
08:20:00 - 08:24:59	20	**2.50**		17	**3.50**
Average	11.2	**5.43**		9.8	**5.90**

4.1.3.2. scenarios

To explore different urban mobility priorities, three distinct weighting scenarios reflect varying perspectives on transportation and sustainability. In each scenario, weights are assigned to a set of key factors: car, bicycle, pedestrian, public transport, road safety, air quality, noise, and equity. These weights indicate the relative importance of each factor within the scenario.

It is important to note that the absolute values of the weights are not directly significant. What matters is their relative proportion—how much more or less important one factor is compared to another. Whether all weights are set to 0 or 10, the final evaluation normalizes the total to 1, ensuring that only the relative differences affect the outcome.

The three scenarios are chosen from the literature review.

Car-Focused: Prioritizes private car use, with all other factors receiving no emphasis.

- car focussed (**car:** 10, **bicycle:** 0, **pedestrian:** 0, **noise:** 0, **equity:** 0, **safety of roads:** 0, **air quality:** 0, **public transport:** 0)

Balanced (traffic management as solution): Assigns equal importance to all factors, reflecting a neutral stance that values all modes and outcomes equally.

- balanced (**car:** 6, **bicycle:** 6, **pedestrian:** 6, **noise:** 6, **equity:** 6, **safety of roads:** 6, **air quality:** 6, **public transport:** 6)

Green Mindset: Emphasizes sustainable and equitable transportation modes and outcomes, with a lower weight for car use.

- green mindset (**car:** 1, **bicycle:** 10, **pedestrian:** 8, **noise:** 7, **equity:** 7, **safety of roads:** 7, **air quality:** 10, **public transport:** 8)

These scenarios illustrate how different values influence transportation evaluation. Since these weights determine the final score of the dashboard, which will be used as input for the global agent, they consequently influence the RL model and the final DQN outcome. So different weights might also produce a completely different RL program.

4.1.3.3. Randomness and different runs

Given that the input for the simulation involves a value for randomness, which is used not only for the OD matrix but also for elements like the car following model, it is crucial that both the old and new scenarios have the same randomness. Comparing entirely different runs could lead to inaccurate conclusions. Due to the lengthy nature of the simulation (especially when reinforcement learning is applied), it has been decided to limit the scenarios to five simulations. The seeds used for the five scenarios are the following: 3,6,9,12,15.

In theory, more runs should be performed but as was mentioned before due to time limitations less runs were performed. The Table 4.3 shows the number of runs needed to determine whether the result is significantly different, indicating that the measure has an effect. The table indicates that many more runs should be permitted to be able to say if real improvements are made. $Cohen'sd$ d is a measure of effect size used to indicate the standardized difference between two means. This metric is widely applied in fields such as psychology, education, and other social sciences to understand the magnitude of differences, helping researchers determine not only the statistical significance, but also the practical significance of their findings. Here's a more detailed explanation of the commonly used values for $Cohen'sd$ [52]:

- A value of 0.2 represents a small effect size.
- A value of 0.5 represents a medium effect size.
- A value of 0.8 represents a large effect size.

Table 4.3: Required sample size for paired t-test at different significance levels and effect sizes ($Cohen'sd$)

Effect size (Cohen's d)	Significance = 0.10	Significance = 0.01
0.2	155	292
0.5	25	47
0.8	10	19

4.1.4. Scoring and visulatisation

An example of network visualization is illustrated in Figure 4.3. The dashboard is designed to provide users with both a high-level overview and a detailed performance breakdown of the transportation network, tailored for quick insights and in-depth analysis.

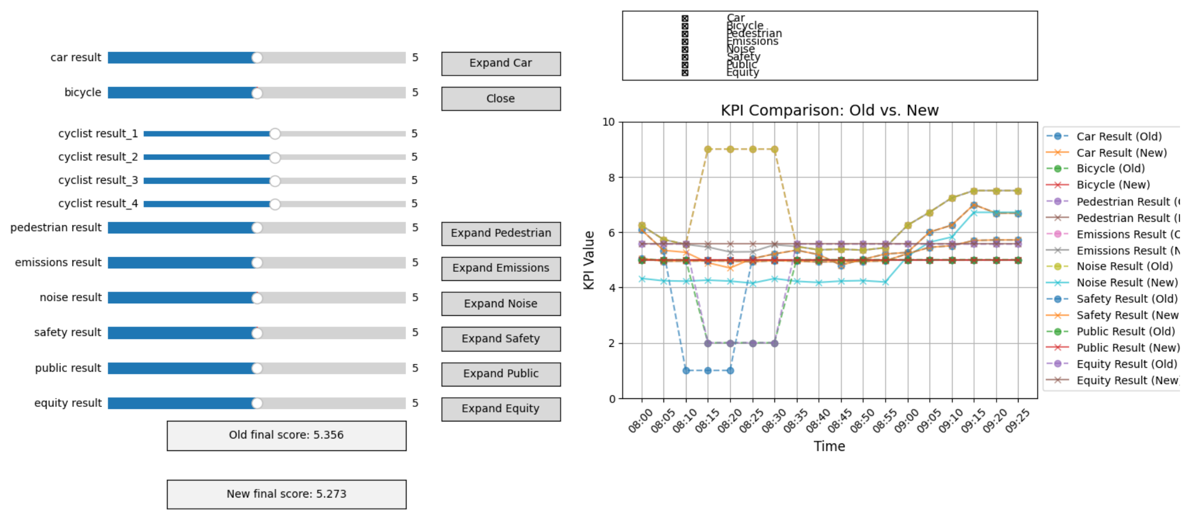


Figure 4.3: Dashboard example

On the left side of the dashboard, sliders are displayed for each major performance theme (e.g., cars, safety, sustainability). These high-level sliders allow users to quickly assess and adjust the relative importance of each theme. This first level of interaction is intentionally simplified to allow quick decision-making and provides an immediate understanding of the overall behavior of the network. It is particularly effective for stakeholders who need a quick, yet informative snapshot of network performance.

Beneath this primary interface lies a second, more detailed level of configuration. Users can expand each theme to view the individual Key Performance Indicators that contribute to it. Within this lower-level interface, users can adjust the weight of each KPI within its parent theme. By default, all KPIs within a theme are assigned equal weights, ensuring a neutral starting point for analysis. However, these weights can be modified to reflect custom policy priorities or scenario-specific goals.

In the bottom left corner of the dashboard, two score boxes are shown: one for the old network and one for the new network. These scores update dynamically as users adjust the theme and KPI weights, providing immediate feedback on how changes in priorities influence overall network performance.

The graphs on the right side of the dashboard display the evolution of network performance over time, segmented by category, and presented in 15-minute intervals. These time-based graphs, especially relevant during peak periods such as the morning rush hour, offer users valuable insight into temporal performance trends.

This comprehensive yet user-friendly structure allows the dashboard to serve as both a strategic tool for high-level planning and a technical interface for detailed scenario tuning. The scoring generated from this dashboard will also be used as input for the RL software discussed in the next section, providing a direct link between policy-driven scoring and adaptive control strategies.

4.2. Graph theory learning

Graph theory is a powerful tool for analyzing and comparing different intersections within a network. Some intersections may score poorly in terms of influence on the overall network, which might not be a major issue. However, even a small improvement at a highly important intersection can have a significant impact on network performance. To evaluate this, Equation 3.1 is used to identify which intersections are traversed by the shortest paths.

4.2.1. Alternative routes

In classical graph theory for transportation problems, analysis typically focuses on the shortest paths in distance. The use in traffic engineering goes way beyond this only approach, but for route choice problems the classical way is using Dijkstra to find the fastest route. However, in traffic engineering, this assumption does not always hold, since other paths might be preferred by a user due to personal preference even if it is not the fastest route. To manage computational complexity, only alternative routes that are no more than 25% longer than the shortest route are considered (the detour factor should be less than 1.25 = alternative route length / shortest route).

The value of 25% was chosen to limit the number of alternative routes, as this can grow rapidly in a given network. Studies show that the average trip length in the Netherlands is approximately 19 km [53], which corresponds to about 15 minutes of driving at 50 km/h.

To estimate the behavior of route choice, a logit model is used with the standard value of $\beta = 0.1$. The probability of choosing a route is given by the following formula:

$$P_i = \frac{e^{U_i}}{\sum_j e^{U_j}} = \frac{e^{-\beta \cdot T_i}}{e^{-\beta \cdot T_1} + e^{-\beta \cdot T_2}}$$

For example, if the main route takes $T_1 = 15$ minutes and the alternative route takes at most $T_2 = 1.25 \times 15 = 18.75$ minutes, the share choosing the main route becomes:

$$P_1 = \frac{1}{1 + e^{-\beta(T_2 - T_1)}} = \frac{1}{1 + e^{-0.1 \cdot (18.75 - 15)}} = \frac{1}{1 + e^{-0.375}} \approx 0.592$$

This calculation shows that even an alternative route that is 25% longer still attracts a significant share of traffic (about 40.8%, with these 2 routes). Therefore, the 25% threshold was chosen to strike a balance between realistic route choice behavior and computational feasibility.

4.2.2. Mathematical formulation link importance

This is addressed through an importance function inspired by the betweenness centrality measure. The function evaluates the number of alternative routes (n) and the total length from origin to destination (l). The variable $x_{o,d}^{n,i}$ is a binary indicator equal to 1 if the route from origin o to destination d passes through intersection i (the intersection of interest) on alternative n; otherwise, it is 0. If an OD pair has only one available route, n can be ignored.

The function first accounts for the difference in length between alternatives and then includes a scaling factor to normalize the average to 1. For example, if there are three alternative routes, each with a different length, shown here: $\frac{120}{300} + \frac{80}{300} + \frac{100}{300} = 1$. Finally, the result is raised to the power of $-3/2$ to emphasize the preference for shorter routes and to reflect that travellers are more likely to choose shorter paths, thus increasing the importance of intersections along those routes. The value estimated to balance since -1 does not do anything and -2 was found to be to extreme therefore -1.5 was used. Further research should examine if the value should be studied in more detail. As an example the split can be calculated using formula 4.1: $\frac{(120/300)^{(-3/2)}}{(120/300)^{(-3/2)} + (80/300)^{(-3/2)} + (100/300)^{(-3/2)}} = 0.24$ the other ones are equal to 0.32 (100) and 0.44 for the shortest route. So when a street has all 3 routes that pass through a link, the value will be 1. When, for example, both routes with lengths 80 and 100 pass through it, the link share will be 0.76 for the route.

$$Impact_{o,d}^{n,i} = \left(\frac{\left(\frac{x_{o,d}^{n,i} \cdot l}{\sum_1^n (x_{o,d}^{n,i} \cdot l)} \right)^{(-3/2)}}{\sum_1^n \left(\frac{x_{o,d}^{n,i} \cdot l}{\sum_1^n (x_{o,d}^{n,i} \cdot l)} \right)^{(-3/2)}} \right), \forall o \in O, \forall d \in D, \forall n \in N, \forall i \in I \tag{4.1}$$

A final consideration is that it is preferable to have multiple routes that use entirely different intersections, rather than routes with significant overlap. In cases of overlap, if one intersection is blocked, all alternative routes (detour factor < 1.25) will also be affected and delayed. Conversely, an alternative route that follows a completely different path increases overall reliability, as it provides a true backup. Since only routes that are no more than 25% longer are considered, any detour should not be longer than the delayed original route. However, when significantly longer routes are included, it becomes necessary to weigh the trade-off between overlap and detour length. This is an important aspect that future research should further investigate. Therefore, each intersection will be assigned a different value for each path. For example, there are 3 alternative routes but only 2 go via a certain intersection, so this will result in $(2/3)^{(3/2)} \approx 0.54$ whilst for the other intersection $(1/3)^{(3/2)} = 0.19$. This factor reduces the importance of an intersection if there are reasonable alternatives, since the sum is not equal to 1 but here equal to 0.73. This shows that more alternatives routes only improves the reliability.

$$b_{o,d}^{n,i} = \left(\left(Impact_{o,d}^{n,i} \right) \cdot \left(\frac{\sum_{n \in N} x_{o,d}^{n,i}}{n} \right)^{(3/2)} \right), \forall o \in O, \forall d \in D, \forall n \in N, \forall i \in I \tag{4.2}$$

When combining and simplifying the formula, the following equation can be found. The output of this formula is a value of the importance of an intersection in a network. Indicating this intersection in a different color allows for a better/deeper understanding of the network to better comprehend where bottlenecks can occur and which intersection are of less importance. Rescaling the final values helps to consider the importance of the KPIs of these different intersections.

$$b^i = \sum_{\substack{o \in O \\ d \in D \\ o \neq d}} \sum_{n \in N} \left(\frac{\left(\frac{x_{o,d}^{n,i} \cdot l}{\sum_1^n (x_{o,d}^{n,i} \cdot l)} \right)^{(-3/2)}}{\sum_1^n \left(\frac{x_{o,d}^{n,i} \cdot l}{\sum_1^n (x_{o,d}^{n,i} \cdot l)} \right)^{(-3/2)}} \right) \cdot \left(\frac{\sum_{n \in N} x_{o,d}^{n,i}}{n} \right)^{(3/2)}, \forall i \in I \tag{4.3}$$

Lastly, as was already mentioned in the literature review, the Python library for networkx allows for efficiently finding the value of connectivity for all the nodes in the Network. This is used on top of the equation 4.3.

For links, the same concept can be utilized as explained for nodes. The more links used for a given path, the more important that link is 4.1. The more different alternative routes there are that use different links, the less important a given link is (see equation 4.2). The only change is that $x_{o,d}^{n,i}$ now refers to an edge and not a node.

4.3. Reinforcement learning

Reinforcement learning will use the final score given by the dashboard as a guiding principle to find the optimal holistic solution. One single RL algorithm can be used to simulate episodes and learn from them. The challenge here is that it could take very long for the software to understand what has impact on what and even more time to effectively tune certain parameters. Learning the localized optimal traffic light policy at intersection level without considering the dynamic impacts among intersection through a holistic overview of the network, likely leads to a suboptimal solution. Therefore, this research proposes the following structure; the structure has a global observer and a local node agent. The reward function in our traffic signal control environment is designed to guide the agent toward efficient traffic management by balancing multiple factors. These include encouraging vehicle throughput, reducing unnecessary signal phase changes, minimizing vehicle waiting, and penalizing overly long green phases.

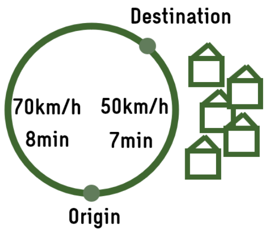


Figure 4.4: global observer is essential

The local agent operates using discrete time steps of one second. Although one second may be relatively long for traffic light control, shorter time frames would significantly slow down the simulation software, and larger time steps would reduce realism. Therefore, a one-second interval is chosen as a practical compromise. Traffic signals remain unchanged within a signal group unless the local agent actively intervenes to alter them. Driver behavior is managed by the microsimulation, ensuring realistic traffic flow dynamics. While the local agent is updated at every time step to enable real-time responsiveness, the global agent is updated less frequently, currently once per simulation run, with the potential to be updated every 15 minutes in future iterations. This setup reflects the distinction between the local agent, which targets immediate operational efficiency, and the global agent, which aligns traffic control with longer-term network-wide policy goals.

The states are measured for all 3 user groups (drivers, cyclists and pedestrians). It is assumed that the factors mentioned above have the greatest impact on the traffic. The local agent will first train on these simple states that can be evaluated every second. The problem is that the local agent will not focus on the broader impacts for the network. Therefore, the global agent will counteract this by looking at this final score when calculating all the different KPIs. An example is shown in Figure 4.4 where the route is shorter via one route, but it might be preferred due to emissions and noise to give more green time to the left turn. Therefore, a positive reward will be given when the local agent chooses that traffic signal group (SG, signal group is a combination of movements). The latter is called ”Signal reward” (see figure 4.5), this could encourage drivers to use that specific routes. This signal reward is only updated once each full simulation, while the other parameters about the traffic state are updated every single time a decision is made (for this research every second).

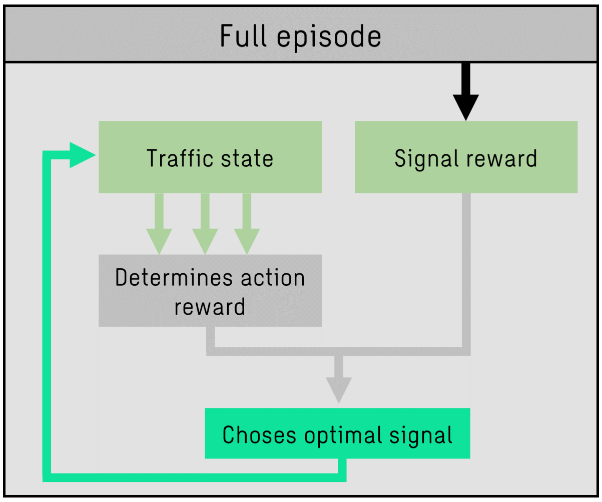


Figure 4.5: Signal reward framework

4.3.1. RL AI structure

As shown in Figure 4.6, a two-level reinforcement learning structure is used. In this setup, the parent or global host acts as an overseer, guiding and coordinating the actions of the local agents. A Q-learning [1] table is not feasible since the solution space is enormously large; therefore, deep reinforcement learning will be used to better deal with the complexity.

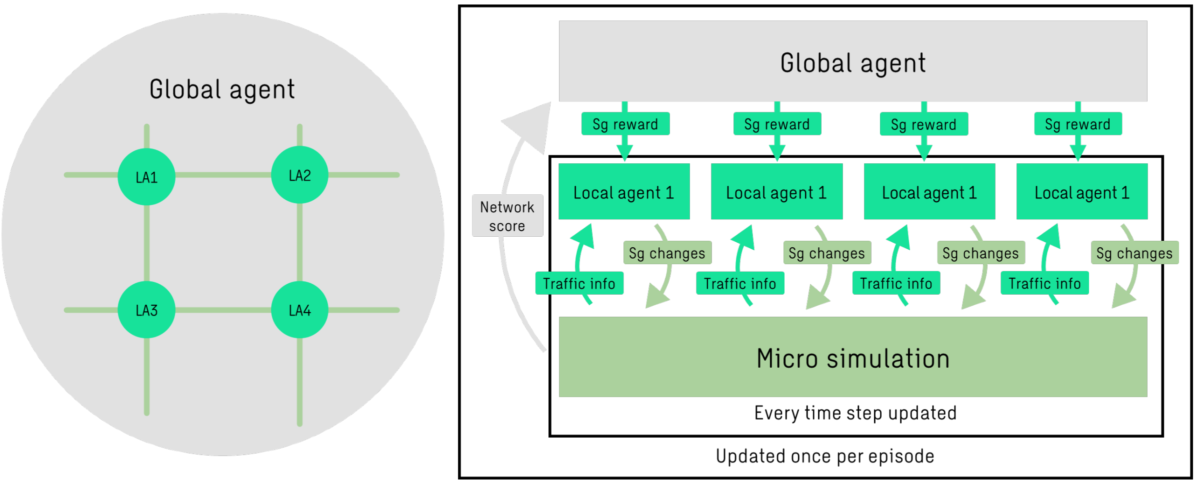


Figure 4.6: Reinforcement learning framework

[1] Q-learning is a value-based reinforcement learning algorithm. The goal of Q-learning is to learn the optimal action-selection policy for an agent interacting with an environment. The agent learns this by updating a table of values (called the Q-table) where each entry represents the value of taking a particular action in a given state.[54]

4.3.2. Local agent

This chapter will focus on the environment and the mathematical formulation of the reinforcement learning algorithm. This chapter will address how the local agent optimizes the traffic flow at intersection level.

4.3.2.1. Environment for local agent

This research assumes that the widespread use of GPS and Bluetooth enabled devices in vehicles allows them to communicate with traffic lights. As a result, traffic lights will receive information when a vehicle approaches within a certain distance (smaller tham 150 meters[2]). However, it is not assumed that these devices provide information about the full travel path of the vehicle. Although this may become feasible in the future, it currently raises significant privacy concerns and is not realistic for the foreseeable future. The data transmitted to the traffic light includes both the vehicle's location and speed. This functionality will be implemented in the simulation in a way that closely mimics real-world conditions, ensuring that the agents trained in the simulated environment can be effectively deployed in outdoor settings.

The intersection includes various road users such as drivers, cyclists, pedestrians, and buses. Traffic light signals are numbered according to the CROW guidelines:

- **Motor vehicle signals:** numbered **1–12**
- **Cyclist signals:** numbered **21–28**
- **Pedestrian signals:** numbered **31–38**

Each group, vehicles, cyclists, pedestrians and buses, is associated with queue lengths, waiting times, and throughput. To reduce complexity, a predefined list of signal groups is used. This are possible combination of traffic lights that will have green at the same time without conflicts. This significantly limits the size of the action space by avoiding the need to consider all possible signal combinations. Furthermore, the maximum intergreen time (the safety buffer between conflicting green phases) is applied between transitions of signal groups.

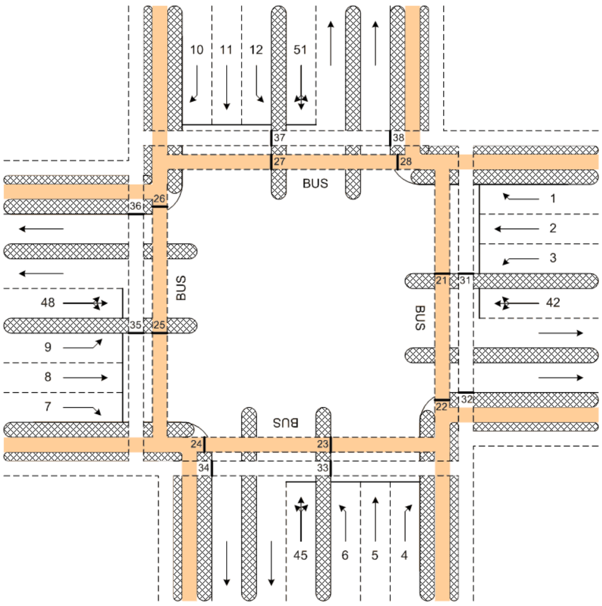


Figure 4.7: Naming of signal groups

[2]A distance of 150 meters is chosen because vehicles typically do not travel faster than 50 km/h when approaching a signalized intersection. At this speed, a car would take approximately 9 seconds to reach the intersection from 150 meters away, giving the traffic light sufficient time to detect the vehicle and respond accordingly. If two traffic lights are closer than 150 meters, the software still functions but has less time to adapt. In such cases, combining the intersections into a single control zone could be more effective and is likely to yield better results.

4.3.2.2. Local agent design and behavior

The local agent is responsible for making real-time traffic control decisions at individual intersections. Its primary objective is to respond adaptively to changing traffic conditions by managing the timing and sequencing of signal groups. Operating on a fine temporal resolution, the local agent evaluates traffic states at each second and decides whether to maintain the current signal phase or switch to another. This decision-making process is informed by both immediate traffic conditions and longer-term learning outcomes. To ensure safety, efficiency, and fairness across all lanes and all road users, the local agent integrates a range of control heuristics, state observations, and reward mechanisms. These components work together to enable dynamic, data-driven signal control that outperforms static or pre-timed traffic light programs.

Action: Every time step, an action is taken to determine whether the traffic light should change. If a change is initiated, the agent also selects which signal group (i.e., traffic movement or direction) to switch to. Following this decision, the traffic signal enters the intergreen phase—a critical safety interval designed to separate conflicting traffic flows during signal transitions.

Summary of mathematical formulation

The following chapter delves into the mathematical foundation of the reinforcement learning (RL) model. The key takeaways are:

Local agent

- A neural network is employed to handle the high-dimensional input data.
- The input consists of a combination of detector occupancy and speed, the currently active traffic light phase, and a timer indicating how long the current phase has been active.
- The reinforcement learning agent learns which actions yield the highest reward and selects actions accordingly.
- The RL algorithm aims to optimize a balance between minimizing vehicle waiting times, avoiding excessively long cycle times, and maximizing traffic throughput (see Equation 4.4).

Global agent

- Sets the signal reward for an entire episode.
- A smart grid search is used to find the parameters that yield the highest reward. This signal reward is a function that scales the reward for the local agent by a factor of 0.5 to 1.5 to adjust the attractiveness of certain actions.
- The solution space of the grid search is twice the number of signal groups. For 4 SG the solution space is therefore a 8 dimension space.

Skip to Chaptre 5

The Q-value function is approximated by a neural network parameterized by θ:

$$Q(s, \text{signal}, \text{time}, a; \theta) \approx \text{DQN}(s, A_{SG}, T_i; \theta)_a$$

Where:

- s: Input state (image tensor of detection and speeds);
- A_{SG}: One-hot encoded signal vector (tensor of length L showing 1 if the lights is green, 0 otherwise);
- T_i: Time feature vector (tensor of length L where each element corresponds to the elapsed time since the start of the intergreen phase for the i-th traffic signal phase);
- a: Action index.

The return is estimated using the n-step return formula:

$$R_t^{(n)} = \sum_{i=0}^{n-1} \gamma^i r_{t+i} + \gamma^n (1 - \text{done}_{t+n}) \max_{a'} Q\big(s_{t+n}, \text{signal}_{t+n}, \text{time}_{t+n}, a'; \theta^-\big)$$

Where:

- γ: Discount factor;
- r_{t+i}: Reward received at time step $t+i$;
- $\text{done}_{t+n} \in \{0,1\}$: Terminal flag (1 if episode ended at time $t+n$, 0 otherwise);
- θ^-: Parameters of the target Q-network;
- $Q(s, \text{signal}, \text{time}, a; \theta^-)$: Q-value predicted by the target network;
- a': Possible actions.

A fixed intergreen period typically consists of the amber (yellow) time and an all-red interval. Its main purpose is to ensure that vehicles from the previous green phase have sufficient time to clear the intersection before vehicles from the next signal group receive a green light. This helps prevent collisions and ensures safe and orderly transitions between phases. Importantly, the intergreen period does not include the minimal green time for the new phase; it is purely a buffer to manage clearance between conflicting traffic streams.

To make better use of the time spent in the intergreen phase, a heuristic is introduced to optimize green time usage. This heuristic aims to maximize the effective green time by aligning the green phase transitions in a way that minimizes lost time and leverages gaps in conflicting traffic flows (as illustrated in Figure 4.8). Initially, the heuristic focuses on maximizing green time within the constraints of the intergreen period itself.

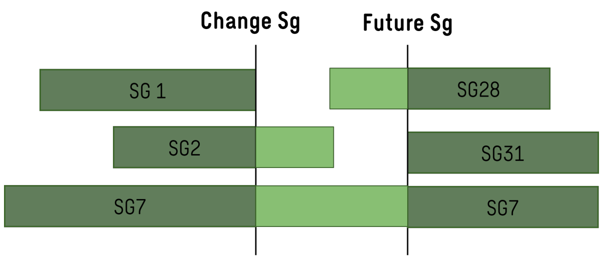


Figure 4.8: Heuristic intergreen maximisation, movement 1 has a conflict with cyclists movement 28 and both movements 1 and 2 conflict with pedestrians crossing in movement 31, Movement 7 is independent of all of them

4.3.2.3. Reward

The reward functions are designed to range between -5 and 5. A value of -5 represents a harsh penalty, while a reward of 5 indicates that the action was considered very good. When a signal switch occurs (i.e., the traffic light changes to a different signal group), the agent receives a reward based on the throughput achieved during that phase. This throughput is normalized by the simulation runtime and adjusted by the number of lanes. The resulting value is then multiplied such that a throughput of one vehicle per second results in a positive reward of approximately 1-2, which is considered a good action (equivalent to 1,200 to 3,200 vehicles per hour, depending on the number of lanes; see [55]).

This normalization helps prevent the agent from always selecting signal groups with more lanes. To balance this, the number of lanes is raised to the power of 1.5: this ensures that more lanes still yield a higher impact, but the influence is less pronounced. If a phase change results in no vehicles passing through, while vehicles in other directions are still waiting, a fixed penalty of -5 is applied to discourage inefficient or premature phase changes.

$$r_1 = \begin{cases} -5, & \text{if } (\sum_i \Delta T_i = 0) \land (\sum_i C_i \neq 0) \\ \dfrac{\sum_i \frac{\Delta T_i}{\tau}}{S^{1.5}} \cdot 4, & \text{otherwise} \end{cases}$$

Where:

- ΔT_i: Number of vehicles that passed (throughput) in lane i ($\sum_i \Delta T_i = 0$: No vehicle passed in this step.).
- C_i: Number of vehicles currently waiting in lane i ($\sum_i C_i \neq 0$: At least one vehicle is waiting.).
- τ: Duration of the current decision step (runtime).
- S: Number of lanes that are turned green.

To promote fairness in lane usage and avoid overload in specific areas, a secondary reward component evaluates the traffic distribution across lanes that are not currently active. This is calculated using a set of predefined lane weights and focuses on lanes that were not given green time in the current decision. The weights are linearly distributed from start to finish to help the software understand that the closer a car is to the traffic light the more certain it can be that the car is in the right lane. Let $S \in \mathbb{R}^{K \times L}$ be the detector data matrix, and $W \in \mathbb{R}^{K \times 1}$ be the weight vector applied to each row of S. Define element-wise multiplication as $S \odot W$, where each row i of S is multiplied by W_i.

Let $A_{\mathrm{SG}} \subseteq \{1, \ldots, L\}$ ($\subseteq$, means subset) denote the set of column indices corresponding to the active signal group lanes.

We define the matrix A as the weighted detector matrix obtained by applying element-wise multiplication of S and W, with the columns indexed by A_{SG} removed:

$$A = (S \odot W)_{\setminus A_{\mathrm{SG}}}$$

The reward component r_2 based on unused detector activity is then given by:

$$r_2 = \frac{N - \sum_{i,j} A_{ij}}{N}$$

Where:

- K: Number of detectors (rows see Figure 4.9), corresponding to positions along a lane (max 20).
- L: Total number of lanes using detectors (columns in the detector matrix shown in Figure 4.9).
- $S \in \{0,1\}^{K \times L}$: Detector matrix indicating usage, where 1 represents an active detector and 0 represents an empty detector.
- $W \in \mathbb{R}^K$: Weight vector applied to each row of S, linearly decreasing from 1.5 to 0.5.

- $A \in \mathbb{R}^{K \times M}$: Weighted detector matrix after removing columns corresponding to the green lanes, where $M = L - |A_{\mathrm{SG}}|$.
- $N = K \times M$: Total number of elements in matrix A.

A penalty has been introduced to prevent traffic signal phases from remaining green for too long. This penalty increases with the duration of the green light but is capped to avoid excessively punishing necessary long phases. It is applied gradually to help the RL agent better understand the trade-offs involved.

For durations between 0 and 100 seconds, a penalty of 0.01t is applied. For durations longer than 100 seconds, the penalty is calculated using the formula 0.045t − 4.4, up to a maximum of 5, which is considered significant. This formula was developed based on insights gained from testing the reward function. The RL agent learns faster if the punishment is applied gradually rather than abruptly after 120 seconds with a fixed value. The maximum penalty of -5 is substantial compared to other rewards and is therefore set as the upper limit.

Although the function is linear, the penalty for durations under 100 seconds is very small, starting at a linear rate between 0 and 0.1 for the first 100 timesteps and then increasing linearly from 0.1 to 5 between 100 and 120 seconds.

$$P = \sum_i \min\left(5.0, \max\left(0.001 \cdot T_i,\ 0.045 \cdot T_i - 4.4\right)\right)$$

Where:

- P: Penalty term representing the cost associated with the duration of signal phases.
- T_i: A vector of length L where each element corresponds to the elapsed time since the start of the intergreen phase for the i-th traffic signal phase.

Additionally, when the agent chooses to extend the current green phase instead of switching to another signal group, a small bonus is added to reward maintaining stable flow when it is beneficial. The value of 3 is used exactly the same as the minimum green time for calculating this reward. This aligns with the actual minimum green time used in the simulation and is intended to prevent the traffic light from constantly switching between phases. Frequent switching could force drivers to brake unnecessarily, reducing efficiency and increasing emissions. Instead, the agent is encouraged to recognize that allowing vehicles to continue driving smoothly, while allowing a few others to wait briefly, is often the better choice.

$$r_3 = \begin{cases} 3, & \text{if swap} = \text{False} \\ 0, & \text{if swap} = \text{True} \end{cases}$$

The total reward function therfore looks like this:

$$r = r_1 + r_2 - P + r_3 \tag{4.4}$$

Where:

- $r_1 =$ Efficiency reward based on throughput and runtime, encouraging smooth vehicle flow.
- $r_2 =$ Reward from weighted detector data reflecting under-utilized lanes.
- $P =$ Penalty proportional to signal phase durations, discouraging overly long or inefficient phases.
- $r_3 =$ Bonus for extending the current green signal without switching, promoting stability.

To avoid switching to signal groups that do not have cars waiting, the timer is reset when there are no cars waiting. This helps software to reduce unnecessary switching. From training it was established that this was most important for intersection with a separate bus lane. Since these lanes see a bus every 10-15 min, which is much longer than the 2 min maximum cycle time.

In general, the reward function integrates throughput efficiency, fairness in lane usage, penalization of poor or excessive phase durations, and strategic encouragement of phase extensions to guide learning toward optimal traffic signal control behavior.

4.3.2.4. Next State / State Transition:

After an action is applied, vehicles move according to the new traffic light phase in Vissim. Queues are updated, and the number of vehicles passing through the intersection is tracked. The traffic state is recorded using 20 detectors placed sequentially from the signal (see figure 4.9). This setup envisions a future where traffic lights and cars will most likely communicate (as mentioned in Chapter 1).

The detectors simulate near-field communication likely to occur between smart cars and smart traffic lights in the future. Each detector is 4.5 meters long, with 2 meters of spacing between them. Given that an average car is approximately 4.5 meters long and typically leaves a gap of about 1 meter in front and behind, this setup effectively captures vehicle presence.

Downstream detectors could be added later to determine if a direction is blocked; by giving green signals to other directions, it would prevent the intersection from being obstructed and avoid dangerous situations. This enhancement could be incorporated into the framework at a later date.

The detectors not only indicate whether they are occupied, but they also record the speed of passing vehicles. These speed and occupancy readings form a matrix that represents the "image" of the traffic state. This method captures the flow of traffic approaching the intersection in high definition. In addition, smart techniques have been implemented to handle diverging lanes more accurately.

As shown in Figure 4.9, point 1 illustrates an intersection where two lanes go straight from north to south. In this scenario, it is assumed that vehicles turning west use the right lane (looking from the driving direction), while those turning east use the other. Therefore, a car in the left lane (looking from the driving direction) will not switch to lane 10.

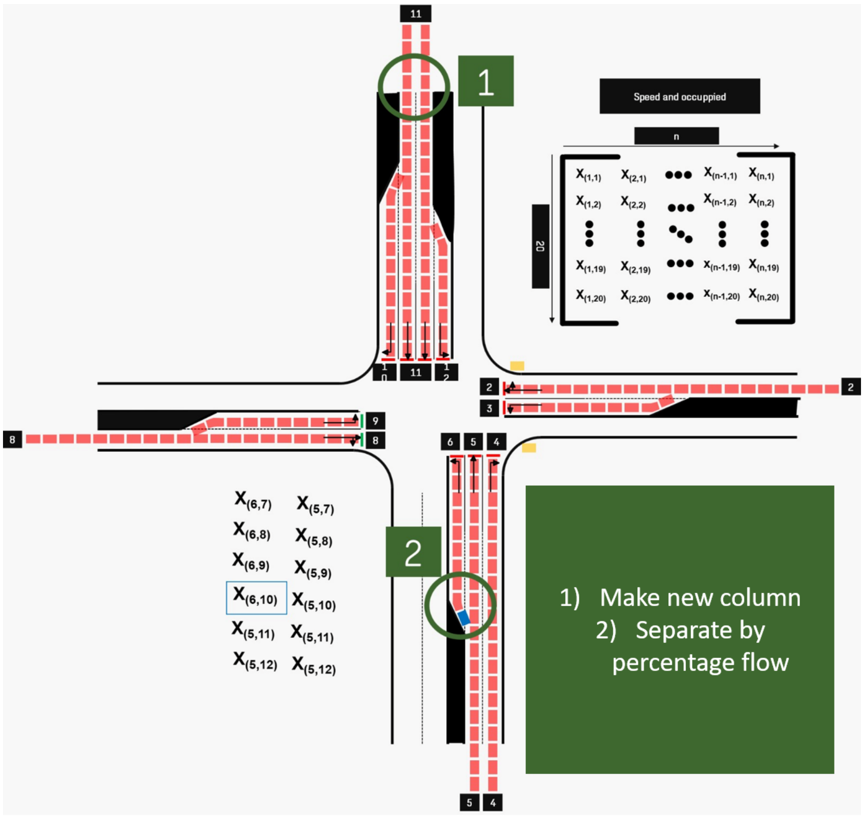


Figure 4.9: State representation local agent, red rectangles show the detectors in the simulation.

Point 2 in the figure shows a situation where 2 lanes diverge. For example, when a vehicle is upstream of the blue detector in lane 5, it can still turn either north or west; this remains uncertain until the vehicle makes a decision. However, by observing turning rate of vehicles in lanes 5 and 6, we can make an educated guess about their intended direction.

To the left of point 2, an array is displayed. This shows that after detector 10 in lane 6, detector 5,11 is used. The occupancy at detector 5,11 is then split according to expected traffic flows, providing a more accurate representation of the state. The speed values are not split as doing so would reduce the actual speed measurements, which would be unrealistic. Only the vehicle's position is considered uncertain when using this virtual queue approach.

4.3.3. Global agent

In addition to localized real-time control, a higher-level agent is introduced to optimize the broader performance of the traffic network (see figures 4.5 and 4.6). This global agent oversees long-term strategy, working above the local agents that operate individual intersections. By monitoring network-wide conditions and aggregating data over time, the global agent is able to detect trends, evaluate policy-level decisions, and adapt to shifting traffic patterns or priorities such as reducing emissions or prioritizing public transport. This hierarchical structure enables a balance between tactical responsiveness and strategic coordination, aligning local actions with overarching mobility goals. The following sections describe the global agent's environment, behavior, and integration with the local agents.

4.3.3.1. Environment for global agent

The global agent is designed to operate at a strategic level, leveraging aggregated performance indicators to influence traffic signal control policies across the network. Rather than interacting directly with real-time signal operations, the global agent evaluates system performance over fixed-length time intervals, currently set at once per episode, although this is subject to further tuning (every 15 minutes would be possibile), to inform its decision-making process.

The global agent's environment is structured to reflect the cumulative behavior of all local agents operating at specific intersections, with performance data aggregated over each episode. This abstraction enables the agent to evaluate broader system-level patterns rather than individual signal decisions. By interacting with this summarized environment, the agent learns to adjust high-level priorities that guide local behavior, creating a feedback loop between strategic intent and operational outcomes. The environment, while simplified at the decision level, retains complexity through delayed, indirect effects and changing traffic dynamics, making it a challenging but realistic domain for reinforcement learning.

4.3.3.2. Global agent design and behavior

Agent's decisions are driven by a set of KPIs that collectively represent multiple dimensions of traffic system performance, such as average vehicle delay, queue lengths, and emissions. These KPIs serve as quantitative measures to evaluate and guide the traffic control strategy.

Each KPI component is assigned a weight reflecting its relative importance in the overall system objective. These weights enable the agent to prioritize different aspects of traffic management, allowing flexible adaptation to changing policy goals. For example, shifting focus from minimizing delays during peak hours to reducing emissions in environmentally sensitive zones.

To optimize these weights, the global agent employs a **grid search method** over a parameter space representing possible weight configurations. Specifically, each KPI weight corresponds to a parameter in a vector α, which scales the Q-values generated by the underlying reinforcement learning model.

By systematically exploring a discretized grid of these weights, the agent evaluates the impact of each parameter combination on the aggregate KPI score through episodic interactions with the traffic environment. After each iteration, the agent identifies the weight vector α^* that yields the best improvement in KPI performance.

Subsequently, the search space is progressively refined by narrowing the range of weights around α^*, with finer grid resolution applied in successive iterations. This **progressive refinement via grid search** allows the agent to zoom in on optimal or near-optimal KPI weightings with increasing precision.

This approach not only supports adaptive tuning of system priorities in response to evolving traffic patterns and policy demands but also maintains interpretability and modularity, as each KPI's influence on the agent's decisions can be explicitly controlled and analyzed.

Let $\mathbf{Q} = [Q_1, Q_2, \ldots, Q_N] \in \mathbb{R}^N$ denote the Q-values produced by the DQN, corresponding to N number of actions (twice the number of SG). The global agent applies a parameterized scaling via a vector $\boldsymbol{\alpha} = [\alpha_1, \alpha_2, \ldots, \alpha_N]$, where:

$$\alpha_i \in [0.5, 1.5] \quad \forall i = 1, 2, \ldots, N$$

The adjusted Q-values are given by element-wise multiplication:

$$\mathbf{Q}' = \mathbf{Q} \odot \boldsymbol{\alpha} = [Q_1 \cdot \alpha_1, Q_2 \cdot \alpha_2, \ldots, Q_N \cdot \alpha_N]$$

The action selection is based on $\mathbf{Q}'$, which indirectly guides lower-level agents.

Let $\boldsymbol{\alpha}^{*(k)} \in [0.5, 1.5]^N$ denote the best parameter vector found at iteration k, and let δ_k be the search half-range for that iteration. The grid at iteration k is defined as:

$$\mathcal{G}^{(k)} = \left\{ \boldsymbol{\alpha} \in [0.5, 1.5]^N \;\middle|\; \alpha_i = \alpha_i^{*(k-1)} + j \cdot \epsilon_k,\ j \in \mathbb{Z},\ |\alpha_i - \alpha_i^{*(k-1)}| \leq \delta_k \right\}$$

where:

- $\delta_k = \eta \cdot \delta_{k-1}$ with $\delta_0 = 0.5$, and $\eta \in (0, 1)$ controls the shrink rate of the search range,
- $\epsilon_k = \eta \cdot \epsilon_{k-1}$ with initial resolution ϵ_0, and $\eta \in (0, 1)$ controls the refinement rate of the grid resolution,

At each iteration, the agent evaluates all $\boldsymbol{\alpha} \in \mathcal{G}^{(k)}$ by cumulative reward:

$$r_t(\boldsymbol{\alpha}) = \text{KPI}_{t+1} - \text{KPI}_t$$

$$\boldsymbol{\alpha}^{*(k)} = \arg \max_{\boldsymbol{\alpha} \in \mathcal{G}^{(k)}} \sum_{t=1}^{T} r_t(\boldsymbol{\alpha})$$

This process continues until a stopping criterion is met (e.g., $\epsilon_k \leq 0.01$), progressively narrowing the search around the best-performing parameter region.

Explanation for the reward definition:

The reward function is defined as the difference between consecutive KPI values:

$$r_t(\boldsymbol{\alpha}) = \text{KPI}_{t+1} - \text{KPI}_t$$

Rather than simply using KPI_{t+1} directly. This design choice emphasizes the incremental improvement in system performance resulting from the agent's actions, rather than the absolute KPI values at isolated time steps. By focusing on the change in KPIs, the agent receives a clearer and more informative feedback signal that reflects how its decisions impact traffic conditions dynamically. This approach helps avoid bias toward states with inherently high KPI values and aligns well with reinforcement learning principles, where rewards represent immediate effects of actions, enabling the agent to optimize for continuous progress rather than static performance snapshots.

4.3.3.3. State Representation

The global agent observes the state at the beginning of each episode, represented as a summary of the performance metrics of the traffic network, specifically the aggregated key performance indicators of the previous episode. This episodic formulation allows the agent to view each interval as a discrete learning instance, capturing the strategic impact of its decisions over time.

Action Space:
The agent's action consists of selecting a vector of parameters $\boldsymbol{\alpha} = [\alpha_1, \alpha_2, \ldots, \alpha_N]$ that scales the Q values associated with the signal groups or timing strategies of N. Each parameter α_i adjusts the corresponding Q-value Q_i, effectively weighting the priority of each signal group. These scaled Q-values guide local agents indirectly by influencing the relative importance of actions, without directly prescribing signal timings.

Reward Function:
The reward at each episode is defined as the change in the overall weighted KPI score, reflecting the difference in traffic performance before and after applying the parameterized Q-value scaling. Formally, the reward is the following.

$$r_t(\boldsymbol{\alpha}) = \mathsf{KPI}_{t+1} - \mathsf{KPI}_t$$

where a positive reward indicates an improvement in system-wide traffic metrics, and a negative reward discourages ineffective parameter settings.

State Transition:
At the end of each episode, the traffic environment is updated according to the cumulative effect of the chosen parameter vector $\boldsymbol{\alpha}$. This results in new traffic states and KPI values that form the state input for the next episode. The agent learns to associate parameter configurations with the resulting performance outcomes, enabling adaptive tuning of signal priorities over time.

Parameter Optimization via Grid Search:
To optimize the parameter vector $\boldsymbol{\alpha}$, the agent performs a progressive grid search within a bounded parameter space (e.g., $\alpha_i \in [0.5, 1.5]$). At each iteration, the agent evaluates a discrete grid of candidate parameters and selects the best-performing vector $\boldsymbol{\alpha}^*$ based on cumulative rewards. Subsequent searches focus on smaller neighborhoods around $\boldsymbol{\alpha}^*$ with a finer resolution, allowing the agent to zoom in on optimal parameter settings.

Design Considerations:
This hierarchical design decouples global strategic tuning from local signal control. The global agent optimizes high-level priorities through scalable Q-value adjustments, while local agents execute fine-grained signal timing based on these priorities. The episodic delayed feedback loop captures aggregated traffic impacts, requiring the agent to learn from system-level trends rather than immediate outcomes. This enables robust adaptation under varying traffic demands and changing KPI weights, maintaining modularity and flexibility throughout the traffic control system.

5
Results and Discussion

This chapter presents the results obtained from the simulation and analysis of the Almelo traffic network. The findings are organized around key performance indicators. Various scoring methods are applied to interpret performance levels in all scenarios. In addition, insights from graph theory and traffic light control strategies are incorporated, including VisVap implementation, to provide a deeper understanding of network dynamics and intersection importance. Lastly, the performance of the KPIs will be compared with the performance when reinforcement learning is used.

The first chapter 5.1 will show the simple network that is used for the OD calibration in Appendix A, and used to show the usefulness of graph theory in more complex networks. The Almelo network is the main network that is used for graph theory and used for RL implementations. The network is simple with six intersections on a 2.5 km stretch of road.

Secondly, understanding the structural and operational dynamics of a road network is crucial for effective traffic management and infrastructure planning. Graph theory provides a robust mathematical framework for analyzing the topological properties of transportation networks, identifying critical nodes and links that play a key role in maintaining overall connectivity and flow. In this chapter 5.2, graph-theoretical techniques are applied to the Almelo road network to determine the relative importance of intersections and road segments based both on their structural position and actual traffic demand. This dual approach helps prioritize interventions in a way that reflects both the network topology and real-world usage.

Third, the training and evaluation of reinforcement learning agents for traffic management at various intersections. We begin with local agent training, where adjustments in reward functions were necessary to address performance issues and avoid overfitting. Specific intersections, K23 and K35, required tailored modifications to address their unique traffic flow complexities.

Next, we assess the performance of the global agent in three scenarios: Car-focused, balanced, and green mindset, highlighting how municipal policies impact traffic optimization (see Figure 5.3.2). While car-focused strategies show significant improvements, green-mindset approaches struggle due to inherent challenges in prioritizing pedestrian and cyclist traffic.

Finally, we compare performance metrics, discussing the trade-offs between efficiency and environmental impact (see Figure 5.4). This chapter emphasizes the importance of a global agent in balancing objectives and ensuring cooperative behavior among local agents for effective traffic management.

Lastly, the dashboard provides geospatial data with time-stamped visualizations, enabling analysis of CO_2 emissions and traffic speeds in different road segments (Figure 5.3.1). Areas with fewer stops show lower emissions, while intersections exhibit higher emissions as a result of frequent acceleration and deceleration. It helps short-term planners, such as traffic operators, monitor real-time speed fluctuations, identify bottlenecks, and evaluate adaptive signal control strategies using reinforcement learning. For long-term planners, such as urban mobility strategists, it allows testing of RL-based traffic optimization strategies to reduce emissions, travel time, and improve network efficiency, supporting informed infrastructure and policy decisions.

5.1. Case study

In the microsimulation, two different networks were created. The first is a test scenario featuring a simple network with four intersections and eight origin-destination (OD) pairs, as shown in Figure 5.1. In this network, the traffic lights operate on a fixed schedule, which simplifies the analysis. The network allows for alternative route choices determined by traffic demand on specific routes. Although easy to construct, this network includes alternative routing options, which is essential to demonstrate the benefits of graph theory. This scenario focuses solely on cars to illustrate the basic principles in action. The simple network is used only to show that graph theory becomes more effective in complex networks (see Appendix E). Due to time constraints, this network was not tested with RL but solely used for OD calibrations and to show that the graph theory formulations are more useful when alternative routes are available. RL was only applied to the Almelo network; therefore, further research is needed to show the methodology works for networks with multiple options; many researchers use a 3x3 network which would be ideal to test the improvements of the current setup.

The second network was chosen because buses are underutilized within this part of Almelo's network. This area contains many bus lines, making it particularly interesting to study. The network includes six intersections and spans approximately 2.5 km, allowing analysis of trip travel times. It is also heavily used during peak hours: in the morning, many people travel into the city, and the reverse occurs in the evening. This segment of the network is multimodal, including pedestrians and cyclists. It is important to note that these two user groups are modeled only near intersections for practical reasons. Modeling cyclists across the entire 2.5 km would be time consuming, as it takes them significantly longer to traverse the network than cars, and the computational load would increase significantly. Since this stretch frequently experiences traffic jams at certain times of the day, there is significant interest in exploring whether and how traffic conditions can be improved.

The second network is the Almelo network (Figure 5.2), and is part of the N road that leads to the city from the east. This stretch of road is part of the collaboration Sweco has with the municipality under the DAES project. Unlike the simple network, the Almelo network provides minimal alternative route options. The network includes buses, pedestrians, and cyclists, adding an additional layer of complexity. The inclusion of the other modes allows the dashboard to be fully utilized. In this network, the traffic lights on the network have been upgraded to allow variable green times per signal group. This means that when heavy traffic approaches, a signal can extend the green times. In addition, traffic lights are designed to skip certain directions if one of the sensors does not detect traffic. More details about these characteristics will be explained in appendix B.

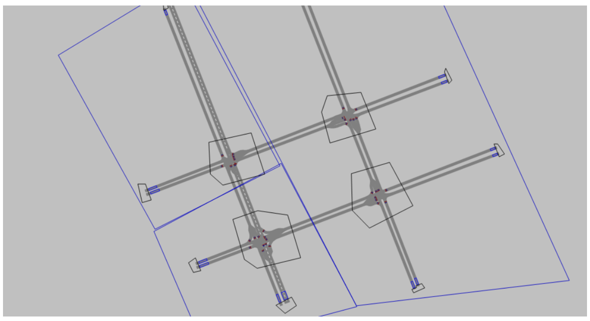


Figure 5.1: Simple network

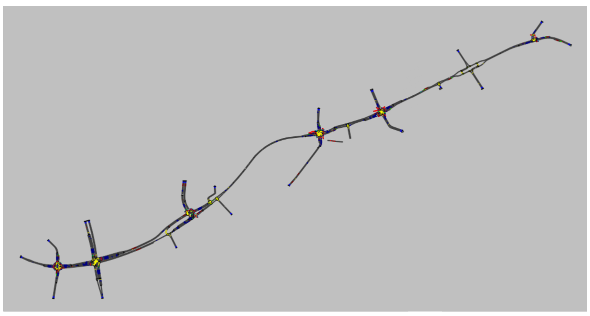


Figure 5.2: Almelo network

As mentioned above, the provincial road from the east that leads to Almelo, known as the N349 road, was modeled using PTV Vissim. This route accommodates traffic from neighbouring towns, as well as a significant volume of vehicles heading towards the highways on the opposite side. Traffic patterns on this link vary notably between peak periods; during the morning peak, there is an influx of commuters travelling into the city, while in the evening, this trend reverses as more individuals leave the city. Consequently, traffic lights that can adapt to these fluctuations in demand have substantial potential to improve traffic flow.

Furthermore, this section of the road passes through an urban area where noise pollution and air quality are major concerns for residents and local authorities. The frequent congestion and vehicle platooning caused by current traffic light configurations not only reduce traffic efficiency but also exacerbate environmental issues such as elevated emissions and noise levels. Addressing these factors is critical to improving both the quality of life in the urban area and the sustainability of the traffic system. A municipal representative noted that current traffic lights contribute to congestion on this main road and the formation of vehicles driving in platoons. For more information on this control structure for the reference Almelo case, again see Appendix B.

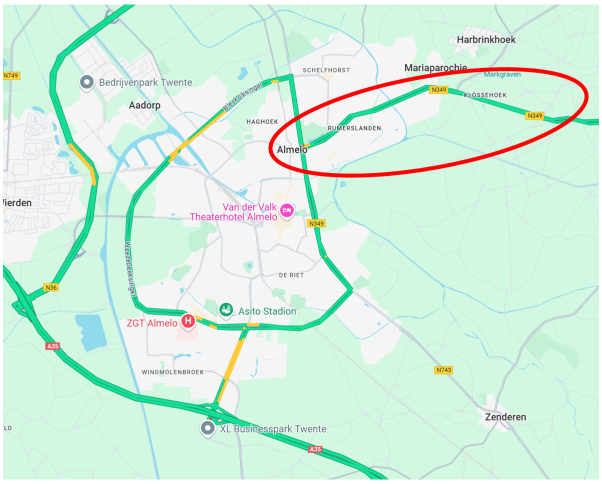


Figure 5.3: Almelo area, red circle shows the N-road that connect neighbouring towns and connects to a small German city

5.2. Graph theory results

This section applies graph theory to the road network of Almelo to identify critical intersections and links based on both their structural role within the network and their operational usage. The analysis begins by evaluating the topological importance of each road segment, measured by the number of origin-destination (OD) pairs that rely on that segment to complete their trips. This provides a structural perspective rooted in graph theory. This part section is not related to policies but purely based on topology and demand. This to make an objective weighting on the network and to adjust the weights so more important crossings will have an higher impact. As mentioned in chapter 4.1 for the slider called Equity the weighting can be updated to adjust for user groups or demography of people living nearby to make it even more complete.

To capture real-world traffic dynamics, this structural score is then combined with traffic demand data. Specifically, each node's importance is calculated using Equation 4.3, and then multiplied by the normalized traffic demand. This results in a combined importance score that highlights road segments which are not only topologically central but also heavily used. These values are derived from observed data and are not based on predefined policy objectives.

Including these importance scores is essential. Without them, the analysis would treat all intersections and road segments equally, potentially underestimating the impact of congestion or failure at highly utilized and structurally central locations. This combined structural-operational approach enables a more accurate identification of bottlenecks and helps prioritize interventions in traffic management and infrastructure planning.

Intersection importance scores were computed using Equation 4.3 and weighted by normalized traffic demand to reflect existing usage levels. Intersections already facing high traffic volumes are thus recognized as being more prone to future congestion. As shown in Figure 5.4 (highlighted in gray), graph theory proves effective in predicting these high-risk areas.

The importance scores also influence the calculation of Key Performance Indicators. Intersections with higher importance contribute more to the overall KPI evaluation. For example, consider Intersection 1 with an importance value of 1.25 and a KPI score (e.g., for car performance) of 4, and Intersection 2 with an importance of 0.75 and a KPI score of 6. The weighted average KPI is calculated as:

$$\text{Weighted KPI} = \frac{(1.25 \times 4) + (0.75 \times 6)}{1.25 + 0.75} = 4.75. \tag{5.1}$$

This approach ensures that performance at more critical intersections has a greater influence on the total network evaluation.

The six key intersections displayed in Figure 5.3 are mapped as a horizontal line in Figure 5.4. Traffic-light-regulated intersections are shown in gray, while other nodes appear in black. The same equation (Equation 4.3) is applied to determine the importance of both nodes and links. This highlights which road segments and intersections are essential for ensuring smooth traffic flow and should therefore be prioritized in operational planning.

5.2.1. Network analysis using only graph theory

As shown in Figure 5.4, all the routes at the end of the network that connect the origin-destination points (1, 2, 3, ..., 19, 20, 21) are equally important, since the same number of OD pairs pass through them. As mentioned above, the center of the network is not very complex, but it does illustrate the underlying logic. In Appendix E, the system with multiple alternative routes is explained to show that it can also deal with more complex networks.

An important consideration in this method is that the selection of the network boundaries significantly influences the results. Specifically, links located on the periphery of the network will generally appear less important than those in the center. Therefore, the placement of the cordon, that is, the delimitation of the study area, has a substantial impact on the analysis. To address the issue that many origin-destination pairs may have negligible actual demand, traffic volumes are incorporated in the next step.

Despite this limitation, the method effectively demonstrates that if a particular link is blocked, a significant number of users will be required to reroute. In cases where no alternative route is available, the calculated importance of the link highlights its criticality within the network.

The Almelo network studied here clearly shows that the central links are the most important and that the farther from the center the links are less important. This is because most OD pairs must pass through the center to connect origins and destinations. The figure below clearly shows that both directions are equally important, as exactly the same number of OD pairs pass through them.

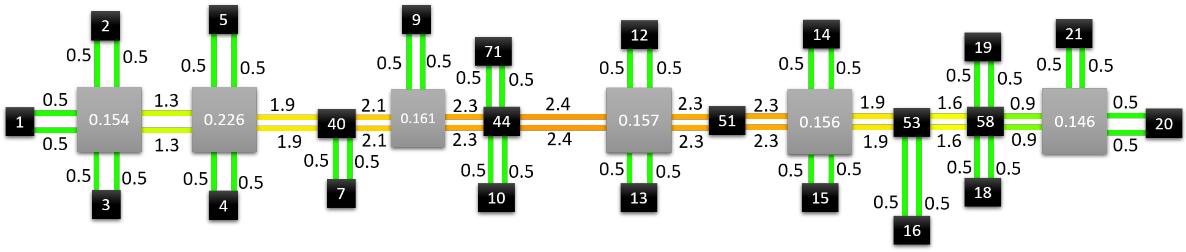


Figure 5.4: Graph theory has been applied to the Almelo network, where the values of the links indicate the number of paths that utilize a specific link (see Formula 4.3). The values for the nodes identify the most important nodes using built-in Python functions (Networkx [45]).

5.2.2. Incorporating traffic demand

Graph theory alone does not provide the full picture, as it is also important to consider how many vehicles actually use a particular road segment. In this case, the demand has been adjusted based on the road segment with the fewest lanes. So the demand is based on segment between 2 nodes with the fewest lanes. This is because once a bottleneck is established, congestion tends to spread more quickly throughout the network. In this way, bottlenecks become even more apparent.

In figure 5.5, it can be seen that a large volume of traffic must travel from right to left in sections with only one lane. In the center of the network, more lanes are available due to the short distances between multiple intersections. However, this presents a limitation: When many lanes congregate in a small area, weaving movements can occur, potentially increasing congestion. This effect is not considered in the current analysis.

However, it is evident that crucial road segments are present both to the left and to the right of node 40, particularly when we consider the number of vehicles that must pass through these segments. Therefore, these areas will require additional attention.

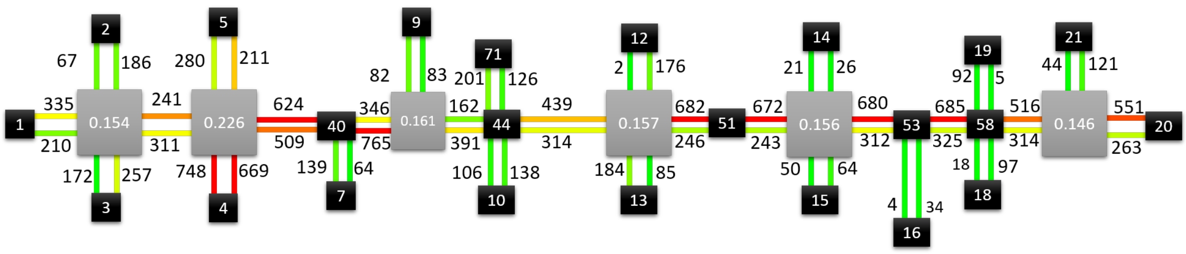


Figure 5.5: Demand on links in Almelo network, vehicle per links in one hour.

5.2.3. Combined analysis and bottlenecks and demand

When both figures (5.4 and 5.5) are combined, a more complete picture emerges. They are combined by multiplying both values for demand and importance value. The figure 5.6 illustrates the difference in traffic intensity between the west-to-east and east-to-west directions. However, it is important to recognize that this trend reverses during the evening rush hour. This means that further analysis is required to account for time-dependent demand patterns and directional traffic flow variations.

The most critical road segments to monitor are those near node 51, which score a relatively high importance value of 4. This indicates that a significant number of origin-destination pairs rely on these links, and its proper functioning is essential to maintaining overall network performance. On the other hand, the least important links in the network score as low as 0.25, which means that only a very small portion of the total traffic demand depends on them.

In addition to node 51, the area surrounding node 40 also contains multiple links that play a crucial role in the network. If these links become congested, there is a high risk of a cascading effect, where traffic jams rapidly propagate to adjacent links. Such gridlock scenarios can severely impact overall traffic flow, particularly in networks with limited alternative routes or low redundancy. A point of criticism for this approach is that the length of the link is not considered while this definitely increases/decreases the change of a gridlock. Further research should append this term to the final Equation 4.3.

It is also worth noting that while the central part of the network tends to have more lanes and intersections. This close proximity (which provides more flexibility in routing) of intersections can also lead to increased stop-and-go waves. These movements can paradoxically contribute to the formation of congestion rather than alleviating it, especially under fluctuating demand conditions. Although this phenomenon is not explicitly considered in the current model, it represents a limitation worth addressing in future studies.

In conclusion, by combining the importance of the graph-theoretic link with adjusted traffic demand data, a more nuanced understanding of network vulnerability and bottleneck formation can be obtained. This methodology not only highlights current critical links, but also sets the stage for evaluating the impact of different traffic scenarios, such as time-of-day effects or infrastructure changes. Such insights are valuable for traffic engineers and urban planners who want to design more resilient and efficient traffic networks.

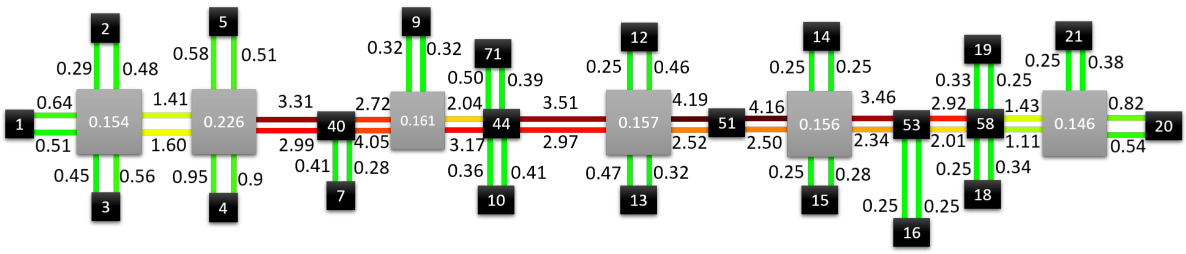


Figure 5.6: Graph theory applied to Almelo network compensated for demand

5.2.4. Bottleneck identification and discussion (simulation)

Increasing traffic volumes in the simulation reveals the initial bottlenecks. These bottlenecks are significantly affected by traffic signal timings. If one direction has fewer green light durations compared to another with equal traffic flow, it is likely to experience congestion more quickly.

The first bottleneck appears in the far west (see Figure 5.7), forming rapidly and becoming difficult to clear once the narrow section is blocked. Near node 51 (Figure 5.7), this can lead to critical situations if queues extend back to the intersection. The fact that graph theory identifies this link as the most important and that it is the first to show congestion supports the method as a good early indicator of network weaknesses.

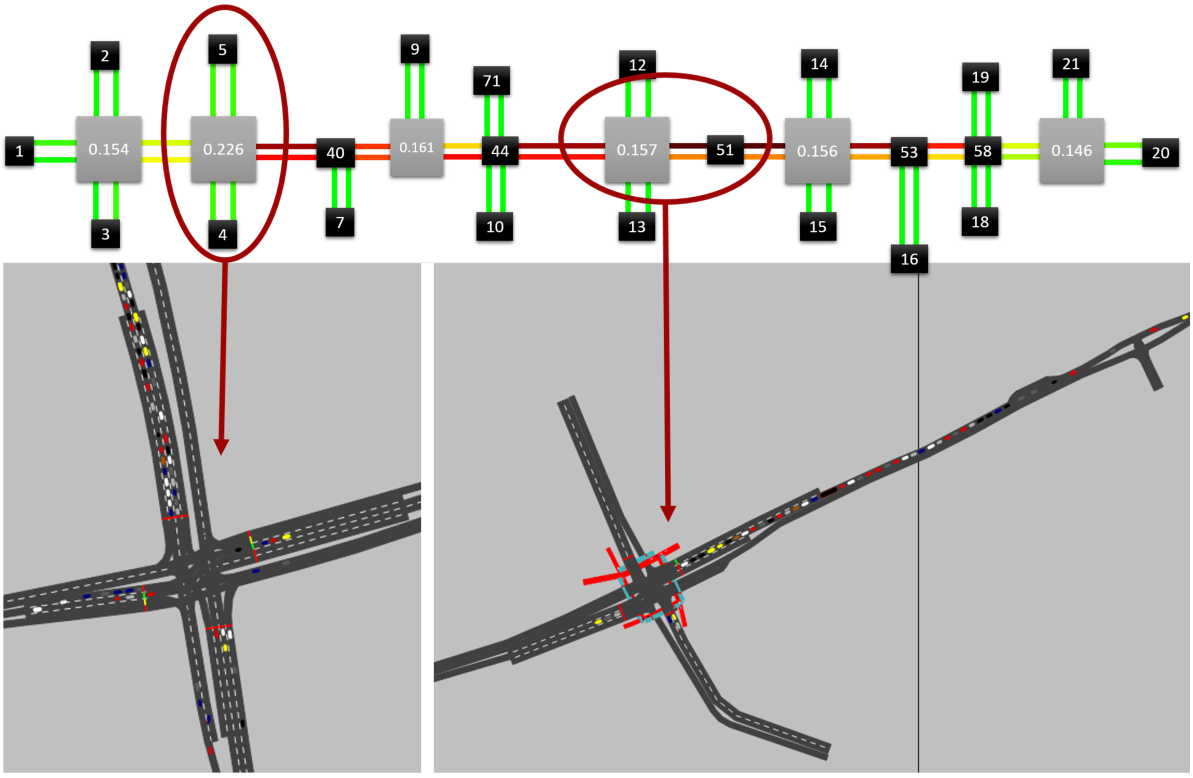


Figure 5.7: Emerging congestion hotspots in the network under increasing traffic demand. The right-hand image represents a lower demand scenario, while the left-hand image shows the network under higher traffic volumes, revealing the development of potential key bottlenecks.

The second bottleneck occurs at intersection 23 (the second gray intersection, with a score of 0.226). While Figure 4.3 suggested that congestion would probably start in the south, it was instead observed from the north. This discrepancy highlights a key limitation of the current methodology near intersections. In future research, it is essential to incorporate the volume of conflicting traffic flows as well as the traffic signal configuration. These factors can significantly influence the effective green time allocated to each direction, and insufficient green time may result in severe delays.

Although both the northern and eastern approaches have four lanes, the northern approach is underutilized, mainly because no vehicles turn left from that direction. As a result, only three of the four lanes are used effectively. In contrast, all four lanes from the east are fully utilized, distributing the traffic load more evenly. This example illustrates a limitation of the current approach.

Further improvements could involve analyzing intersections in more detail, such as considering how many lanes serve each direction and the number of vehicles per movement. With this addition, the method could predict the congestion more accurately at intersections. Also incorporating the distance between intersection would make the tool better in predicting problems. Since a blocked intersection will give back spills forming a gridlock which is rather hard to solve and takes a lot of time before this unwinds.

5.3. Almelo network results

The training was performed on different machines to speed up computation time. Initially, each model was trained individually, which took between a couple hours to around 36 hours for the most complex intersection. The average speed decision was made is once every 0.0648 seconds which means that a simulation of 1.5 hours takes approximate 5 minutes to run. Later, the intersections will be combined. The final step is to train the global agent. Due to the limited time and substantial computing power required, although a grid search is proposed, a few options are tested to evaluate whether this concept is worth pursuing in future research.

5.3.1. Local agent training results (intersection Almelo)

All local agents were initially trained using the same reward function. However, minor adjustments were necessary for the two-intersection scenario to improve performance. Figure 5.8 shows the learning curves for both intersections. The left-hand graph corresponds to a simpler intersection, which converges more quickly to a satisfactory policy. In contrast, the right-hand graph depicts a more complex intersection that exhibits a slower learning rate and signs of overfitting in later training stages, evident in the downward trend of the overall performance score.

Overfitting[1] was not observed across all intersections but occurred primarily in cases with highly unbalanced traffic flows or where one approach experienced minimal traffic. Prolonged training in these situations led the agent to over-optimize for dominant traffic patterns, effectively ”forgetting” to allocate sufficient attention to underutilized lanes. This highlights the importance of balancing training duration and reward function sensitivity for intersections with asymmetric or sparse traffic conditions.

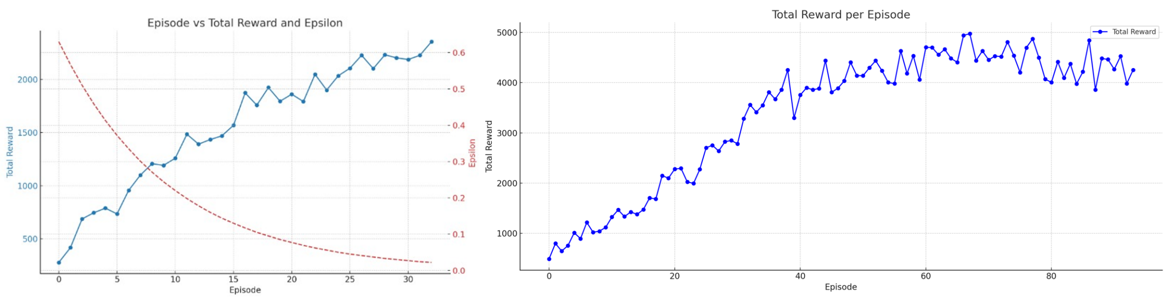


Figure 5.8: Reward improvements during training are correlated with decreased randomness. The red line represents the reduction in randomness per run, while the blue line indicates the total reward gained by the RL program. The upward trend of the blue line demonstrates that the agent is consistently pursuing actions that are deemed beneficial and yield high rewards.

5.3.1.1. Training results

Figure 5.9 shows the locations of the two intersections used for detailed evaluation. Intersection 35 is the first major bottleneck in the network, as identified in the graph theory analysis. Intersection 34 was selected as a comparison point because it initially performed well, making it interesting to assess the extent of potential improvements through reinforcement learning.

[1] Overfitting occurs when the software has trained excessively on specific datasets, resulting in difficulties adapting to new scenarios. Consequently, it learns actions that may be beneficial in certain situations but are generally considered incorrect. This is evident when the reward function decreases, indicating that the software is pursuing overly restrictive actions.

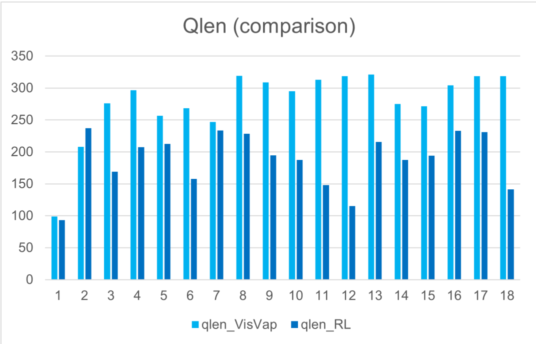
Figure 5.9: Locations of intersections used in local agent training evaluation.

At Intersection 35, the most apparent and impactful improvement is illustrated in Figure 5.10a, where queue lengths are substantially reduced across nearly all time intervals. This reduction indicates that the local agent has learned to manage green time more effectively and distribute it in a way that minimizes build-up. In several instances, the average queue length decreased by more than 50%, a clear testament to the agent's proficiency in optimizing traffic flow.

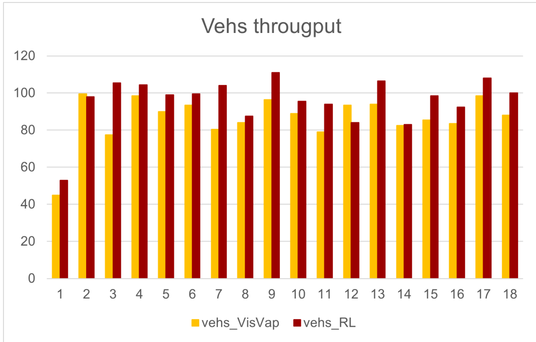
These reductions in queue length yield notable downstream effects on environmental performance. As shown in Figures 5.10c and 5.10d, there are meaningful drops in emissions under the RL-controlled policy compared to the baseline. The emissions of CO, NOX, and VOCs exhibit significant decreases, underscoring the environmental benefits of the optimized signaling approach.

Interestingly, an increase in throughput (see Figure 5.10b), although seemingly small, is visible at Intersection 35. During each timestep, there is an increase of around 10 vehicles processed, which is significant in the long run. This incremental rise plays a crucial role in explaining the observed lower emissions, as fewer stop-and-go waves result in more efficient traffic flow and reduced idling. Consequently, the local agent's optimization in queue management and emission reduction becomes evident despite the intersection operating near its physical capacity.

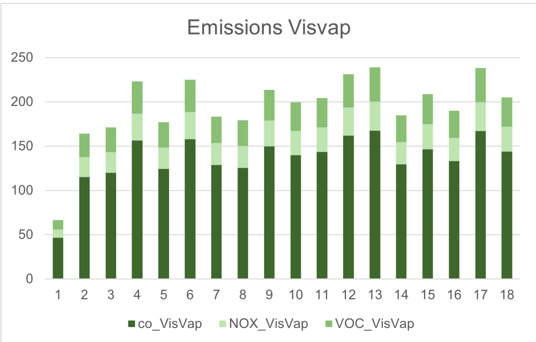
However, even in a capacity-constrained environment, the agent demonstrates an ability to make more intelligent phase-switching decisions. By allocating green time in a balanced manner that considers the needs of all approaches, the agent effectively minimizes vehicle idling and optimizes secondary performance indicators. This shows the agent's capacity to enhance secondary KPIs such as queue length and emissions, even when the primary metric of throughput hits a plateau, thereby significantly improving the overall efficiency and sustainability of traffic operations at Intersection 35.



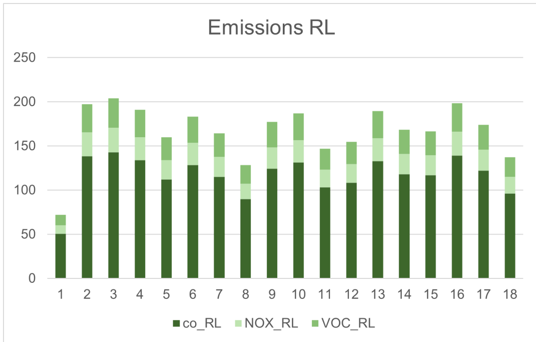
(a) Queue length comparison



(b) Vehicle throughput



(c) Baseline emissions (VisVAP)



(d) Emissions after RL training

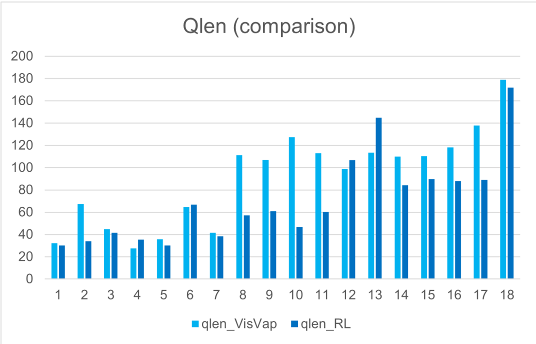
Figure 5.10: Performance comparison of the local agent vs. baseline for Intersection 35, including queue lengths, throughput, and emissions.

At Intersection 34, the situation is in stark contrast to that of Intersection 35. Figure 5.11b demonstrates that the intersection is not yet operating at full capacity, which enables greater flexibility to improve overall traffic throughput. This untapped capacity allowed the reinforcement learning agent to significantly increase the number of vehicles processed during the simulation. This is not what happened since there is enough spare capacity that all vehicles still go through although more optimised flow still reduce pollution and waiting times.

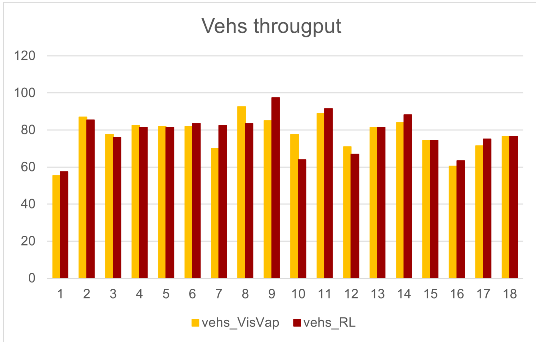
In addition to small throughput gains, Figure 5.11a indicates that queue lengths are generally reduced across all intervals, which is encouraging for traffic management. However, the presence of two pronounced spikes in the queue length trajectory reveals moments where the agent made less effective decisions. These temporary inefficiencies suggest that the agent's policy is not yet fully stable and would benefit from additional training epochs to better handle edge cases or infrequent traffic patterns. Still, this is a good result and underpins that even though all cars are processed, the way cars are processed can still affect the waiting times.

Despite these anomalies, the overall performance trend remains positive. The smoother flow achieved under the agent's policy contributes to a reduction in stop-and-go waves, which are known to significantly increase emissions due to frequent acceleration and deceleration. This improvement in flow efficiency is corroborated by Figures 5.11c and 5.11d, where both baseline and RL-generated emissions are compared. The agent successfully demonstrates its ability to improve environmental sustainability by reducing emissions while simultaneously minimizing delays.

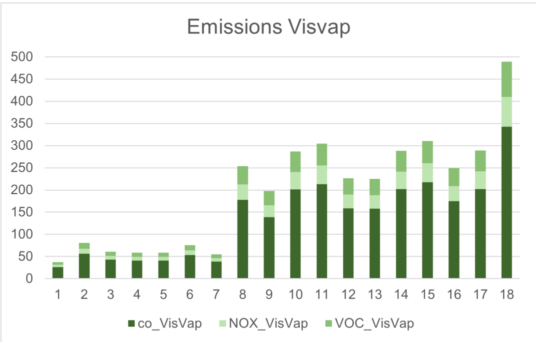
Furthermore, although the original system manages to process all vehicles, the data suggest there is room for significant improvement in reducing queue lengths, highlighting the effectiveness of the RL agent in optimizing traffic management strategies at Intersection 34. This adaptability and potential for refinement make the reinforcement learning approach a powerful tool for future traffic solutions, capable of tackling diverse and dynamic traffic situations.



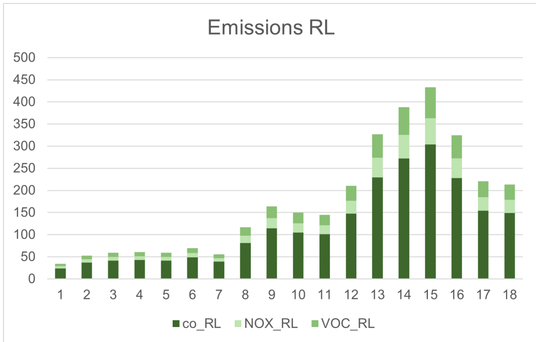
(a) Queue length comparison



(b) Vehicle throughput



(c) Baseline emissions (VisVAP)



(d) Emissions after RL training

Figure 5.11: Performance comparison of the local agent vs. baseline for Intersection 34, showing queue lengths, throughput, and emissions.

It is important to note that not all intersections exhibited improvements as significant as those shown above. In cases where traffic inputs were highly imbalanced, such as one or two approaches receiving the vast majority of vehicles, reinforcement learning agents sometimes struggled to generalize effectively across all phases. Despite these challenges, the overall results remain highly promising. Even under complex and uneven traffic conditions, the trained local agents demonstrated the potential to enhance traffic flow efficiency and reduce emissions, making this approach a strong candidate for real-world deployment.

5.3.1.2. Key insights gained during training

During the training process of the Deep Q-Network agent, several valuable insights were discovered. These findings provide a deeper understanding of the dynamics involved in training reinforcement learning models for traffic control systems.

One of the most prominent insights is that the fewer signal groups there are, the faster the agent is able to learn. This is a logical outcome, as the complexity of the decision space is reduced with fewer groups, resulting in a more efficient learning process. However, increasing the number of signal stages from four to five leads to a disproportionately longer training time, exceeding 135% of the original duration. This increase is attributed to the expanded state and action space, which significantly complicates the learning task for the agent.

Another important observation is that DQNs struggle with large negative values. This becomes particularly problematic when the magnitude of these values is extreme. Similarly, extremely large positive values can introduce significant instability into the learning process. Therefore, it is critical to normalize reward values to ensure numerical stability and maintain effective learning performance.

Given that a single simulation can take between 5 to 10 minutes depending on the device used, it is essential to optimize the training process for efficiency. Reducing the number of unnecessary or random actions helps in accelerating convergence and makes better use of computational resources.

During early training episodes, when epsilon (ε, the exploration rate)is still high, the local agent tends to take random actions the entire time. These random decisions often lead to illogical behavior, which can cause traffic congestion and even gridlock situations. Once a gridlock occurs, it becomes extremely difficult for the agent to learn effectively due to the lack of actionable feedback.

To address this, it is essential to gradually increase traffic demand in proportion to the decrease in randomness (ε). This ensures that the agent is not overwhelmed during early training phases and that learning remains possible and effective. Preventing complete system standstills is critical for allowing the agent to evaluate the consequences of its actions correctly.

The discount factor (γ) plays a vital role in helping the agent evaluate the long-term impact of its actions. In reinforcement learning, gamma determines how far into the future the consequences of a current action are considered. For traffic control applications, this becomes especially important because the state of traffic can change significantly from one moment to the next.

Based on the experimental results, a discount factor between 0.8 and 0.85 was found to be ideal for the current setup. This value strikes a balance between considering the long-term impact of actions and maintaining the relevance of immediate outcomes. For instance, with $\gamma = 0.8$, an event occurring 11 steps in the future retains approximately 10% of its original influence, which aligns well with the decision-making needs of the agent in this environment. A higher γ could lead to an overestimation of long-term rewards, potentially destabilizing learning in environments with high variability, such as intersections with dynamic traffic patterns.

It is important to note that the optimal choice of γ is closely tied to how far into the future the agent can effectively "see," which in turn depends on the physical characteristics of the traffic system, specifically the placement of detectors and vehicle speeds. In this experiment, the detector range was 120 meters, and vehicles were assumed to travel at 50 km/h. At this speed, a vehicle would be detected approximately 9 seconds before reaching the traffic light if it did not decelerate. Given that the reinforcement learning agent operates with a time step of 1 second, using a γ that effectively looks 11 steps (or 11 seconds) ahead is consistent with the observable environment.

However, if the detector range were extended or if vehicle speeds differed significantly, the discount factor would need to be adjusted accordingly. Additionally, if the time step was shortened, for example, from 1 to 0.1 seconds, the agent would need to look over 100 steps ahead to cover the same 9-second planning horizon. Thus, the choice of γ must be carefully calibrated to reflect the temporal and spatial resolution of the traffic system in which the agent operates.

5.3.1.3. Reward function adjustments for specific Intersections

Finally, the reward function was slightly modified for two specific intersections. It is no coincidence that both of these intersections were also discussed in the chapter on graph theory, as they represent structurally complex cases within the traffic network.

The first intersection, referred to as K23, is characterized by two highly dominant traffic flows (see white and black arrows in Figure 5.12). Due to the larger number of lanes in one direction (particularly the north-south corridor), there is naturally more throughput on that axis. Although the model corrects for the number of lanes, the structural imbalance in the intersection remains difficult to fully neutralize. Since there is a lot of traffic east-south but not as much south-east, making this hard for the program to interpret.

A significant portion of vehicles travel east-west and vice versa, especially on the east side. This imbalance frequently results in long queues forming. Once a queue reaches the single-lane segment near the intersection, the situation rapidly deteriorates, causing a bottleneck that the agent fails to resolve efficiently.

To address this, an additional penalty was introduced to the reward function whenever a vehicle was detected standing still on the last detector (furthest upstream). This change prevents the formation of queues beyond the detection zone. Without this penalty, the neural network had learned that queues behind the detectors were not penalized, and thus not relevant to the reward. As long as visible queues (within the detection range) were cleared, the agent would receive a high reward, even if there was significant congestion just outside its perception range.

The adjusted reward function ensures that such backlogs are now included in the optimization process, improving the agent's ability to manage traffic more holistically at this intersection.

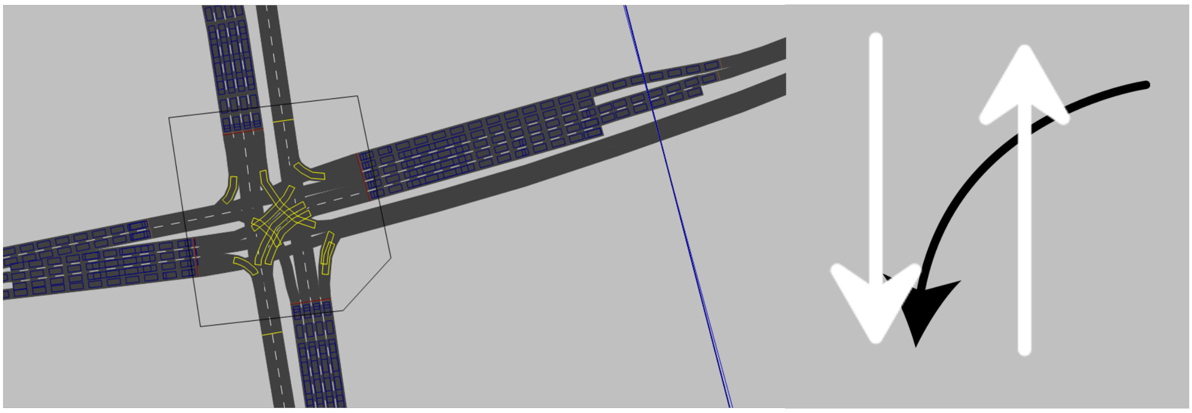


Figure 5.12: Modified reward structure for intersection K23 to prevent queues from extending beyond detection zones

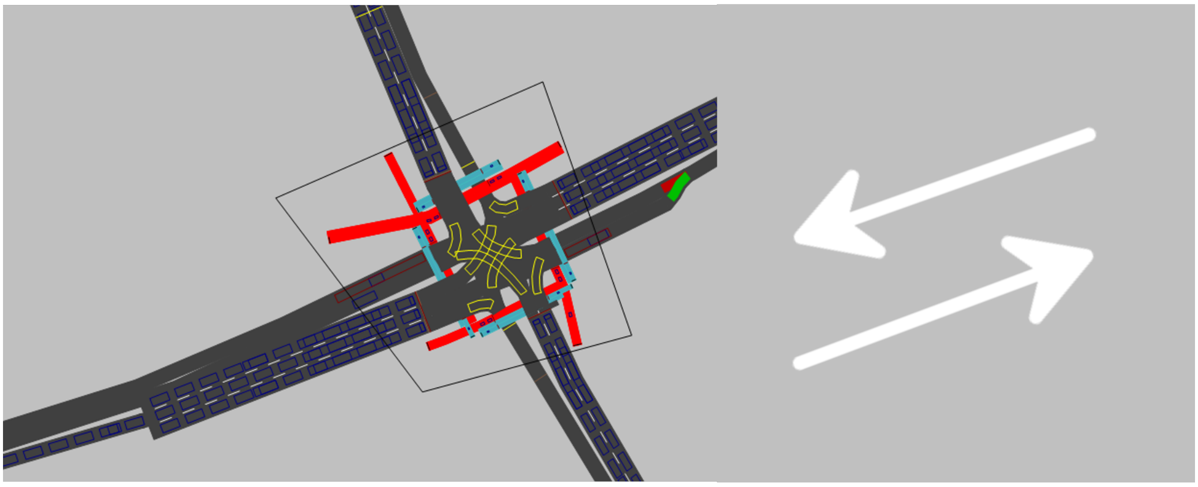


Figure 5.13: Reinforced reward for the dominant east-west flow at intersection K35 to prevent counterproductive switching.

The second intersection, K35, experiences a high volume of traffic moving in the east-west direction. This intersection suffered from a similar issue to K23, where queues would extend back into a single-lane road segment, effectively blocking upstream traffic and making recovery very difficult.

The core of the issue lies in the fact that very few vehicles travel from west to east. As a result, the reward associated with the main east-west movement was underestimated by the DQN. Consequently, the traffic light frequently switched to allow non-dominant flows (such as north-south), even when the demand was relatively low. This frequent switching led to interruptions in the more critical east-west flow, eventually resulting in severe congestion.

To resolve this, an additional positive reward was assigned to the signal group responsible for the east-west movement. This adjustment helped the agent better learn the importance of maintaining flow in this dominant direction, reducing unnecessary interruptions and improving overall traffic efficiency.

5.3.2. Global agent results (network Almelo)

In this section, the performance of the global reinforcement learning agent is evaluated in three distinct transportation scenarios - Car Focused, Balanced, and Green Mindset - within the Almelo network. These scenarios reflect different prioritizations of mobility factors such as vehicle flow, environmental impact, and equity among road users. By comparing the performance of the global agent with the baseline network configuration (VisVap), this analysis highlights the critical role of a global agent in balancing competing objectives and avoiding tunnel vision in traffic optimization. The section further explores the impacts of various policy adjustments implemented by the agent, providing insights into their effectiveness and trade-offs in real-world traffic management.

5.3.2.1. Scenario descriptions and weights

It is important to note that the car-focused scenario presented here is not intended to reflect a realistic or desirable traffic management strategy. Instead, it serves solely as a testing case to examine the behavior of the system under extreme conditions. For example, assigning zero weight to cyclists and pedestrians can lead to highly unrealistic outcomes, such as these modes of transport never receiving a green signal, an outcome that would be unacceptable in any real-world application.

The purpose of this approach is to test the limits of the methodology and to underscore the importance of a more holistic perspective. As previously explained, the three different scenarios, Car-Focused, Balanced, and Green Mindset, have been designed to demonstrate the necessity of employing a global agent. Without a global view, an agent may become narrowly focused on optimizing a single objective (such as minimizing car delay), leading to tunnel vision and suboptimal outcomes for the overall system. Each of these scenarios will be discussed in detail in the following section.

Table 5.1: Comparison of transportation mindsets across key factors

Factor	Car Focused	Balanced	Green Mindset
Car	10	6	1
Bicycle	0	6	10
Pedestrian	0	6	8
Noise	0	6	7
Equity	0	6	7
Safety of Roads	0	6	7
Air Quality	0	6	10
Public Transport	0	6	8

5.3.2.2. Baseline performance of the old network

All three scenarios will be evaluated against a baseline scenario developed using VisVap, which represents the original network configuration. As outlined in the Method chapter, the VisVap output is scored on a scale from 2.5 to 7.5 for each time step using Method 3. 10 means that the score is optimal given the original situation and zero means that the KPI worsened significantly. This scoring approach enables a consistent evaluation of each individual scenario, Car-Focused, Balanced, and Green Mindset, allowing for a direct comparison of their performance over time and highlighting the impact of different traffic management strategies.

When evaluating the performance of the old network scenario, the following results were observed (see table below (5.2)). It is evident that this network performs best in a Car-Focused mindset, scoring 5.73, which is higher than 5.34 and 5.27 for the other policies. Almelo has a Car-Focused mindset, and the street is designed as a major artery into the city. This alignment with expectations is not surprising, as the network was originally designed as a connector road, prioritizing vehicle flow. Consequently, when more weight is given to factors such as noise or pedestrian-friendliness, the network score will be lower.

The Green Mindset scores the worst, primarily because the car, which is the best-performing KPI, only accounts for 1.7 percent of the final given this policies (Table 5.1). As figure 5.14 illustrates, pedestrians and cyclists score the worst; therefore, they drag down the overall score. When they are given more weight in the green scenario, they lower the overall score. This logical result demonstrates that even a single aggregated score can offer meaningful insight into network performance.

Table 5.2: Comparison of transportation mindsets with average final scores, the higher the score the better the score is with a maximum of 10

Scenario	Car Focused	Balanced	Green Mindset
VisVap	5.73	5.34	5.27

Looking at the simulation in more detail reveals several noteworthy patterns (see Figures 5.14). One of the clearest trends appears in the emissions score, which shows a gradual deterioration over the course of the morning period (Figure 5.14b). This outcome is expected: the simulation starts at 08:00 when the network is still relatively empty, and as more vehicles enter the network, especially between 08:10 and 08:25, the emissions increase accordingly. Importantly, this increase is not due to higher average speeds, which typically correlate with more aggressive driving and thus greater emissions. Instead, it is the increased volume of cars that leads to higher cumulative emissions.

Between 09:00 and 09:20, an interesting interaction emerges between the car, cyclist, and pedestrian scores (Figure 5.14a). During this period, as car performance lowers, there is a notable increase in cyclist and pedestrian scores. This inverse relationship suggests that the traffic light logic prioritizes certain user groups at the expense of others. Specifically, when signal timings are adjusted to favor cars (e.g., longer green phases for motor vehicles), vulnerable road users, such as cyclists and pedestrians, are subjected to longer waits or more frequent interruptions. This trade-off becomes apparent when car scores drop after 09:00, coinciding with one of the highest points in pedestrian performance.

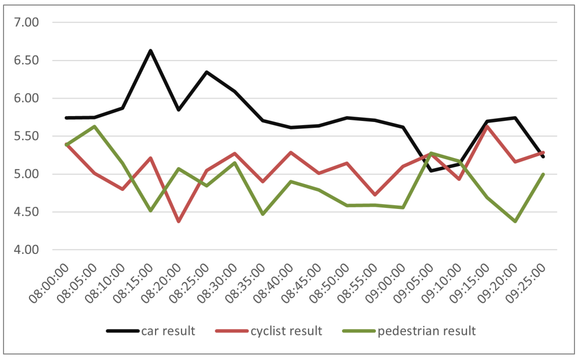
The noise and emissions results (Figure 5.14b) are somewhat inversely related, although not in a straightforward way. Normally, one can expect noise levels to increase with speed, whereas emissions can vary depending on acceleration, stop-and-go behavior, and engine efficiency. In this case, the noise levels actually show a slight improvement over time, likely due to the reduced speeds caused by the growth of congestion. However, the expected sharp decline in noise does not occur, probably because the increase in vehicle volume offsets the quieter conditions created by slower traffic. This suggests a complex interplay where increased congestion reduces speed-related noise but amplifies the overall ambient noise due to sheer volume.

Figure 5.14c shows that the equity score remains perfectly flat throughout the simulation. This is by design: All intersections were preconfigured to give priority to emergency vehicles and buses. Since this configuration is uniform across the network and does not change during the simulation, the equity metric is constant and offers little additional insight under these conditions.

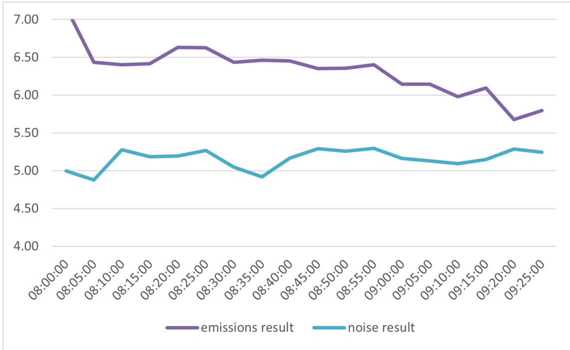
The public transport performance score, however, tells a very different story. Its highly volatile behavior, featuring sharp peaks and valleys, can be explained by the infrequency of bus arrivals. At times when no bus is present on the network, the metric appears favorable. But when a bus does appear, its performance significantly drags down the score, likely due to delays, interference from other traffic, or unfavorable signal timing. This lack of consistent bus service creates an unstable performance profile.

The safety score presents a less intuitive pattern. A steep decline is visible between 08:00 and 08:07, likely attributable to an early-phase shift as vehicles begin to populate the network. As traffic stabilizes, the safety metric levels off quickly and remains relatively stable for the remainder of the simulation. This suggests that after an initial adjustment period, variations in speed and interactions between modes of transport become limited, resulting in fewer fluctuations in perceived or modelled safety conditions. Higher fluctuations in this metric was expected but could not be seen.

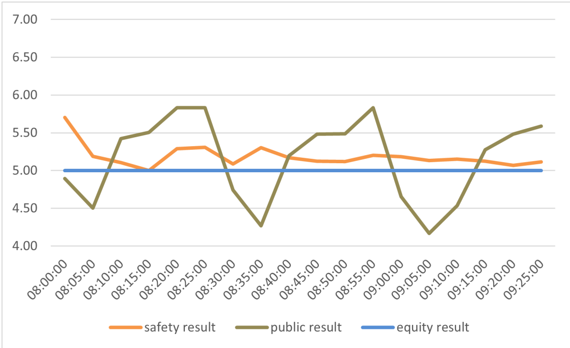
It is important to note that the results presented are based on a limited number of simulation runs (5), which may affect the clarity and consistency of certain performance indicators. For example, the safety metric, which is inherently sensitive to rare events and localized interactions, may not exhibit a clear or stable trend when only a few runs are aggregated. Since safety results can vary significantly depending on small changes in vehicle behavior, pedestrian movements, or signal timing, a limited dataset may obscure underlying patterns or exaggerate short-term fluctuations. Consequently, some of the observed stability or variability in the scores (Figure 5.14c) should be interpreted with caution, as it could be influenced more by sampling limitations than by the actual dynamics of the system. Additional simulation iterations would help smooth out these inconsistencies and provide a more robust understanding of safety performance across the network.



(a) Performance of various transport modalities: car, cyclist, and pedestrian.



(b) Impact on emissions and noise levels in the old scenario through time.



(c) Assessment of safety, public infrastructure, and equity goals in the old scenario.

Figure 5.14: Combined results from the old scenario across different metrics.

5.3.2.3. Impact of global agent policies on performance

The three columns in Table 5.3 represent different municipal policy mindsets: Car-Focused, Balanced, and Green Mindset. These mindsets reflect the priorities assigned by municipalities through weighted values. The rows represent different signal control strategies implemented by the global reinforcement learning (RL) agent, which adapts traffic light timing based on various assumptions about policy objectives. These strategies test how well the network performs under different implementations, depending on the municipality's selected priorities.

The VisVap row shows the baseline performance for each policy scenario, using the original network configuration without adaptive reinforcement learning. The three RL-based strategies: Car First Policies, Only RL, and More Swaps demonstrate how different control logics impact network performance.

In the Car-Focused mindset, the RL agent significantly improves the score, particularly when applying the Car First policy, achieving the highest value (6.43). This indicates that when the agent's behavior is closely aligned with the municipality's priorities, it can effectively optimize traffic performance.

In the Balanced scenario, the results are mixed. The Car First policy yields a slight improvement over VisVap (5.34 vs 5.32), while other strategies perform similarly or slightly worse. This suggests that even in a balanced context, policies favoring smoother car flow may indirectly benefit broader indicators like noise, emissions, and safety, due to reduced stopping.

The Green Mindset scenario yields the most unexpected outcome. All RL strategies underperform compared to the VisVap baseline. Even though pedestrian scores show marginal improvement, the overall scores decrease. This is largely due to the high volume of cars in the network, which pass through multiple intersections and heavily influence global KPIs. The current reward functions used by the RL agent still prioritize vehicular flow, and thus struggle to adapt effectively to a pedestrian- and cyclist-oriented policy.

It is important to note that the chosen RL strategies were based on an educated guess of what might perform well under each policy mindset. Conducting a full grid search to comprehensively tune the global agent would have been prohibitively time consuming. Therefore, the underperformance of certain strategies, particularly under the Green Mindset, likely stems not from a fundamental flaw in the RL approach but from the difficulty of correctly anticipating optimal control behavior in complex systems. This highlights that while traffic management strategies may seem straightforward at a conceptual level, their underlying dynamics are far more intricate and challenging to grasp without systematic tuning. The fact that the car policy does show real improvements does indicate that the global agent does perform it functions which give confidence that improvements are possible for the green mindset with a more intricate update RL strategy.

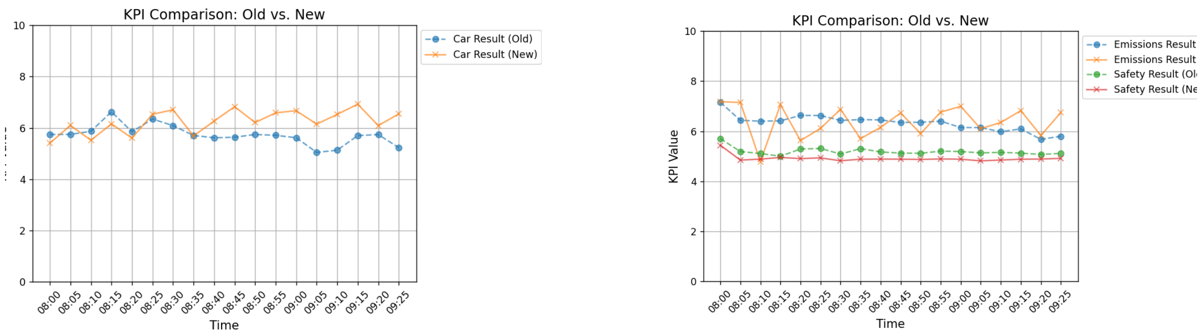
These findings underscore the importance of carefully aligning the agent's reward functions with the intended municipal policy goals. Future research should explore how a global agent can be better adapted to reflect more equitable traffic management priorities, ensuring fair consideration of all road users.

Table 5.3: Comparison of transportation mindsets with average final scores

Scenario	Car-Focused	Balanced	Green Mindset
VisVap	5.73	5.32	**5.25**
Car first policies (RL)	**6.43**	**5.34**	5.19
Only RL	6.25	5.31	5.19
More swaps (RL)	6.11	5.28	5.17

5.3.2.4. Performance metrics and trade-offs

The figure below shows that there is a performance boost when evaluating the system using RL. An increase in traffic performance of cars can be seen. However, this comes at the cost of a slight increase in noise pollution (although the change is minimal). It is also evident that emissions increase. A decrease in safety can also be observed, due to the proximity of intersections and the greater variation in vehicle speeds, caused by cars that need to brake more frequently when approaching red traffic lights. These trends clearly indicate that reinforcement-based traffic lights perform better when focused solely on car-related performance metrics. Cyclists, in particular, are negatively affected, as they often experience longer waiting times due to the longer cycle lengths in the updated traffic signal strategy.



(a) Car Results: OLD vs RL

(b) Safety Noise Results: OLD vs RL

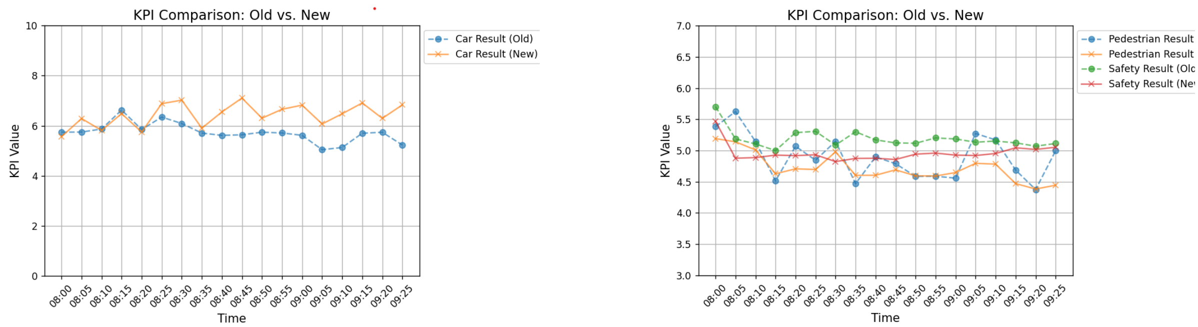
Figure 5.15: Comparison of OLD vs RL methods across car results and safety noise performance.

The Balanced scenario shows only a minimal decline in performance, with the overall score dropping slightly from 5.34 to 5.31. This reinforces the idea that the mother agent is essential for balancing the objectives of the local agents and mitigating the selfish behavior they may otherwise exhibit. For the other two scenarios, the concept of the global agent has been elaborated upon to show that it indeed leads to an improvement in performance scores.

5.3.2.5. Policy-specific observations

In the Car-Focused scenario, a deliberate design choice was made to provide additional rewards to vehicles traveling along the main corridor. Due to time constraints, a full training cycle could not be completed. Instead, a shortcut was used: the reward signal for the relevant signal groups was increased by 5 to 10%, making these signal phases more favorable. While this approach does not replace a complete training process, it serves as a reasonable approximation to simulate the intended behavior. As shown in Figure 5.16a, the network maintains a steady performance level, indicating that congestion is effectively avoided.

As seen in Table 5.3, the green policies are not particularly effective. Frequent switching between signal phases results in vehicles stopping more often, which explains the significant fluctuations in emissions. Safety also declines due to increased variations in vehicle speeds. Furthermore, because the VISSIM model lacks long stretches of dedicated cycling paths and pedestrian walkways, the impact on pedestrians is relatively limited. The complex behavior of cyclist and pedestrian platooning is not well captured in the current setup, reducing the insightfulness of the related indicators. Further research is needed to better model and evaluate these aspects.



(a) car performance OLD vs RL with car policy

(b) Green Mindset Improvement

Figure 5.16: Comparison between old network and green-optimized network

In fact, it appears that the global agent will be an essential component in further research, especially when considering the broader aspects of mobility. The global agent, by focusing on coordination and making adjustments where necessary, encourages thinking in terms of a collaborative system involving various key components (in this case, intersections). In addition, the global agent ensures that long-term goals are maintained by providing insight into the dynamics of the complex network. The global agent training will take longer, so more efficient tools will need to be used. The result do show that the global agent is important to keep the egoistic behavior of the local agent in check to improve cooperation.

5.4. Key insight of the visualization of the dashboard

The developed tool provides geospatial data output, enabling more visual inspection of the results. This geospatial data is also time-stamped, allowing changes over time to be analyzed. Figure 5.17 shows the emissions per segment of the lane. As expected, road segments where cars can drive without frequent stops, such as those without traffic lights, tend to have lower emissions and, therefore, appear in lighter colors. In contrast, intersections involve frequent acceleration and deceleration, which explains the higher emissions observed there. As discussed multiple times in this research, the morning peak traffic demand is mainly directed from east to west, which is also reflected in the figure below.

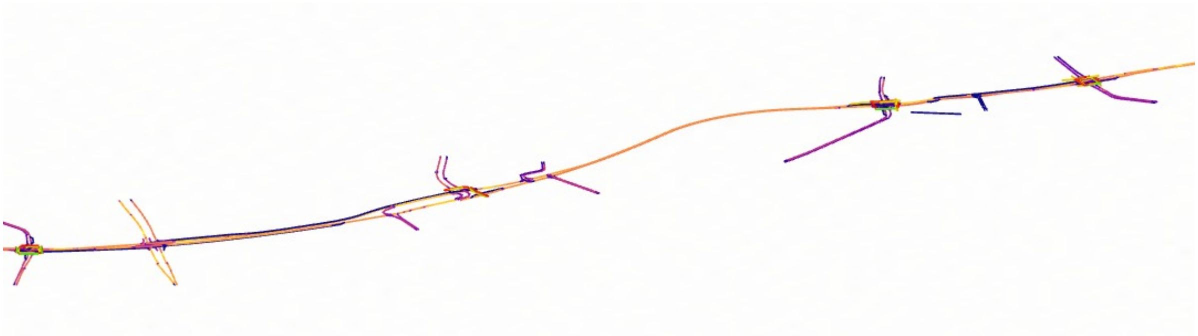


Figure 5.17: CO_2 emissions per lane segment, darker colours means more emissions and yellow means less emissions

The figure below displays the speed of each segment of the link over time, offering valuable information on how traffic conditions evolve throughout the day. This dynamic visualization is especially useful for both short- and long-term planners.

For short-term policy makers, such as traffic operators and incident managers, the ability to observe speed fluctuations in near-real-time provides a clear view of where and when bottlenecks occur. By simulating RL-based traffic control strategies under varying demand conditions, they can assess how well the system adapts to real-time fluctuations. This helps answer the question of how RL-based simulations can mitigate delays and improve traffic flow at the intersection level. The interactive nature of the tool, including features such as time window sliders, allows these planners to pinpoint critical time windows, identify the effects of sudden demand spikes, and evaluate the effectiveness of adaptive signal control strategies in real-world scenarios.

For long-term planners, such as urban mobility strategists and infrastructure investment decision makers, the tool offers a platform to test how various RL-based traffic light optimization strategies perform across different planning horizons and which policies changes outperform others. By simulating future traffic demand patterns, long-term planners can explore the impact of alternative traffic (signal) policies on broader goals, such as reducing emissions, reducing average travel time and achieving network efficiency. The visual and time-sensitive nature of the data allows users to track performance trends, understand the systemic impacts of localized decisions, and ultimately design traffic signal strategies that align with long-term urban mobility and sustainability objectives.

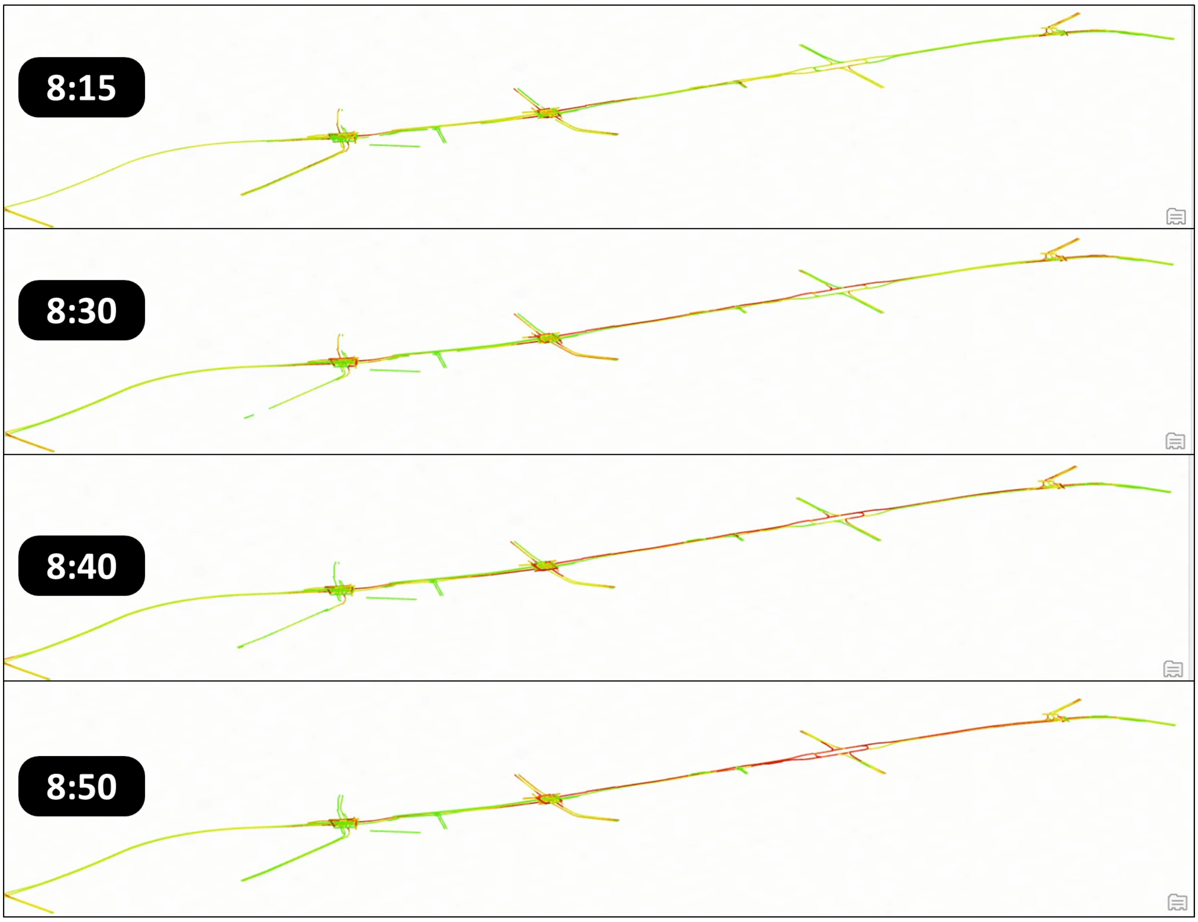


Figure 5.18: Speed on each segment, formation of a queue

In summary, this simulation and visualization tool bridges the gap between operational traffic management and strategic transportation planning. It empowers short-term planners to respond effectively to current traffic conditions and equips long-term planners with data-driven insights to inform infrastructure and policy decisions aligned with future mobility goals.

6 Conclusion

In this chapter, the main results and takeaways will be discussed and highlighted. The various topics range from dashboarding the network using KPIs to applying graph theory to identify bottlenecks, and utilizing RL-DQN (Artificial Intelligence) to enhance the traffic light control structure. The primary research tools used were Python and PTV Vissim. The dashboard is a prototype intended to demonstrate the functionality of the final tool but is not yet aesthetically pleasing or user-friendly.

This research covers a wide range of topics, touching on them all while expanding the scope of graph theory and reinforcement learning to consider agents that take a holistic system view. Because multiple elements were integrated into a single study, not all tools were validated with sufficient precision. For example, there are questions about RL's performance when fewer or faulty detectors are used. In graph theory, calculating all routes in a complex graph can be time-consuming, and more shortcuts will likely be necessary.

Although these concerns are valid, the research successfully demonstrates the importance of a holistic view and the significance of the tools used for further exploration in this field. Graph theory provides valuable information on which intersections and segments are more prone to spillback, indicating that improvements in these areas can enhance the overall network. The dashboard, coupled with RL, highlights the importance of visualizing results to better understand them. The sliders help align the RL model with the policies set by the municipality. This research shows that the alignment between policies and outcomes is intertwined and should be managed as such.

The final remark is that VisVap was used as the base scenario for this research. For future studies, Secol (a program used by smart traffic lights in the Netherlands) should be incorporated into Vissim to check if the RL model can still outperform the traditional traffic light system.

6.1. Summary of key findings

The study effectively combines graph theory and reinforcement learning to optimize traffic networks. Graph theory helps identify critical intersections and bottlenecks, especially when paired with traffic demand data. However, it falls short in capturing complex traffic dynamics such as weaving sections and the influence of green light distribution at intersections. The latter can affect which directions experience delays, and the former can be the origin of traffic jams, as the software struggles to understand that even though there are two lanes, in practice, it often functions as a single lane due to vehicles shifting from one lane to another. Incorporating such complexities will significantly improve predictability.

Reinforcement learning combined with a neural network shows promise for adaptive traffic control. Training local agent separately immensely increases computation speed. The training time is very dependent on the number of lanes and signal groups. To get a well balanced and stable DQN it is key to have a reward function that has small values and is positive. The latter helps to speed up the process of finding good actions. A discount factor between 0.8-0.85 is essential for the reinforcement model to think ahead but not too far so that it can learn from the current traffic state since it can switch very fast. Tailored reward function adjustments were crucial for complex intersections to prevent issues like hidden queues (after reach of detectors) and to overprioritize dominant traffic flows. Hidden queues occur when spill-back happens and intersection upstream gets blocked, so the queue at that specific intersection improves although there is more congestion in the system as a whole.

The global agent outperformed scenarios with only local agents active in 2 out of 3 cases. In the third scenario, the performance was identical. The improvements in the successful scenarios ranged from 0.2 to 0.03, which may seem small but are significant when considering the scale of the values used; these improvements are larger than 3% for the lowest increase. It's important to note that the full potential of the global agent has not yet been reached, so bigger gains can be expected. Further research should explore this potential more thoroughly.

This study highlights the critical role of a global agent for network-wide optimization. While local agents focus solely on situations near their respective intersections, they often overlook the bigger picture. The global agent can counteract this narrow approach by considering the complexities observed by the local agents. Local agents do not have overarching policy goals in mind and will primarily aim to minimize waiting times, which might conflict with the municipality's desired outcomes.

RL, particularly using DQN, shows promise for adaptive traffic control. For local agents, waiting times were reduced by approximately 20% to 35%, and overall emissions were decreased by more than 10%. These improvements are already impressive, and when applied to an entire network such as Almelo with 30 traffic lights, the positive impact could be substantial.

Training local agents revealed that learning speed is closely tied to signal group complexity, and stable training requires careful reward normalization and optimal discount factors. Adjusting reward functions specifically tailored to complex intersections was crucial to prevent issues like hidden queues and to prioritize dominant traffic flows. The reward function works for most general cases, but as highlighted in Chapter 5, some intersections with slight imbalances or frequent lane changes may need specific alterations.

Useful updates to the reward function include lowering the impact of lanes with high traffic volumes so RL does not solely focus on these directions, and giving more weight to cars close to the stop line, as lane changes are less likely and certainty increases. Adjusting the latter influences how effectively the traffic light responds to incoming cars, potentially reducing stops or decelerations.

The research emphasizes the critical role of a global agent for network-wide optimization. Although a car-focused RL approach improved car performance, it negatively impacted other KPIs, such as emissions and safety. In contrast, a Balanced scenario, guided by a global agent, maintained performance with minimal trade-offs. This highlights that a global agent is essential for balancing diverse policy goals and preventing local agents from acting selfishly, ultimately ensuring that long-term, holistic mobility objectives are met. This methodology shows that the optimum depends on which policy themes are given more priority, so different weights can results in a totally different optimum.

Short-term RL reaction to demand: By combining graph theory with a digital twin simulation, traffic managers can proactively pinpoint weak spots in the Almelo network before congestion or other issues even arise. The time window in which one could look ahead ranges from couple of minutes to a few days. since the tool can be expanded to forecast demand given historical data using current trends. This foresight is critical for municipalities in minimizing public nuisance and keeping traffic flowing smoothly. The RL models further enhance this by being trainable on various worse-case scenarios, such as major event traffic or sudden influxes from incidents. This equips the system with robust strategies to manage high-demand situations in real-time, ensuring the municipality can react effectively and minimize disruption to daily life.

Translating KPIs into traffic light control: Through a global RL agent that intelligently coordinates all local intersection agents throughout the network which is more in line with the policy goals. This is where municipal policy truly comes to life. The dashboard serves as a vital tool, providing traffic managers with real-time insight into the network's current performance against various policy-driven KPIs (e.g., air quality, bicycle safety, public transport efficiency, and reduced noise).

Crucially, this dashboard combined with the simulation allows municipalities to:

- **Model future scenarios:** Visualize how the network would perform if specific policies were enacted, for example, prioritizing public transport, encouraging cycling, or reducing emissions.
- **Evaluate policy impact:** Quantify the potential benefits and trade-offs of different policy choices before costly physical implementation. This helps municipalities make data-driven decisions that align with their long-term urban development and sustainability goals.
- **Adapt to evolving goals:** The RL system can be re-trained as the goals of municipal policy evolve, ensuring that the traffic management system remains aligned with the vision of the city for a more livable and efficient Almelo.

The municipality should make/have a micro simulation of their network, and all agents should be individually trained (if RL will be applied), which will take time depending on the network. But once training is completed, traffic management transforms from a reactive exercise to a powerful and proactive policy-shaping tool for the municipality.

6.2. Limitations of the research and further research

One of the biggest challenges faced in this research is the long computation times, which make it nearly impossible to scale the approach effectively. Although training a single intersection can be completed in a day, combining multiple intersections and training them together results in exponentially longer processing times. This is mainly due to the communication time between VISSIM and Python (Figure 6.1 and Table 6.1): As the number of intersections increases, the frequency and volume of communication also increases, significantly slowing the simulation. Zooming out more means also simulating more vehicles (including pedestrians and cyclists), which only slows things down further. As a result, training intersections simultaneously becomes impractical. Apart from using the COM interface[1], other approaches were also considered for system integration. One can also opt for Visvap or other tools, but they have a limited number of options. Visvap for example only allows upto 4 different structures, which reduces the software to adapt and limits its usability. COM, on the other hand, allows for way more flexibility, but it does not allow one to set all intersections at once. Since the speed of simulation scales linearly with the intersections, the problem becomes worse the more intersections are present.

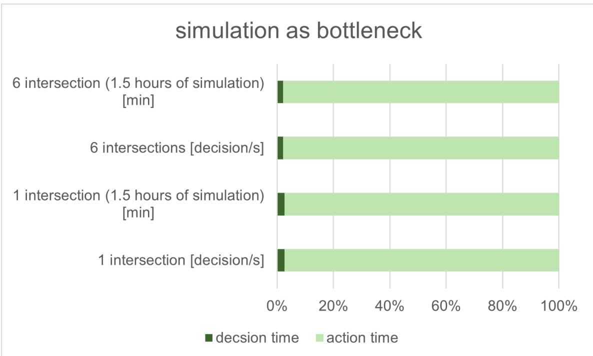


Figure 6.1: Simulation shown to be the bottleneck

Table 6.1: Performance metrics for intersection simulations (1 vs. 6 Intersections over 1.5 Hours)

Metric	**1: [decision/s]**	**1: (1.5h) [min]**	**6: [decision/s]**	**6: (1.5h) [min]**
Decision time	0.001394	0.125503	0.00775	0.6975
Action time	0.052425	4.718218	0.3485	31.365
Storing time	0.010981	0.988279	0.00465	0.4185
Total time	**0.0648**	**5.832**	**0.3609**	**32.481**

[1]The component object model (COM) is a Microsoft-developed platform that enables interprocess communication and dynamic object creation. It allows software components to interact regardless of the programming language used.

One of the main limitations of the current approach is the long computation time, which poses a significant challenge for real-time traffic responsiveness. In the case of Almelo, a relatively small city, even the existing training duration becomes infeasible (around 30 traffic lights). Each time an adjustment is made to the network, the global agent must be retrained from scratch, and any modified local intersections require their respective agents to be retrained as well. This makes the tool impractical for frequent updates or dynamic traffic environments.

Ideally, the full training time for the global agent should be reduced to under a week (finish entire grid search). This target reflects the realistic need to adapt to policy changes, which can occur regularly. Once training is completed, traffic lights should be able to respond to day-to-day traffic fluctuations, provided that these do not deviate drastically from training conditions. However, when infrastructure changes occur, such as the addition of new intersections or newly approved housing developments, traffic patterns can change significantly. In these cases, faster training of multiple local agents becomes essential to maintain effective traffic control.

Furthermore, from a traffic controller's perspective, the ability to simulate faster than in real time is crucial. To anticipate future conditions and evaluate different scenarios, a simulation speed of at least 10x real time would be necessary. This would enable predictive modeling, including best- and worst-case traffic scenarios, allowing proactive adjustments before problems arise. Unfortunately, tools like Vissim, when combined with COM interfaces, are too slow for large-scale networks and cannot support this level of real-time forecasting and control. Therefore, the advise is to use different tools; many researchers have used SUMO instead of Vissim for training many local agents (see literature).

Another limitation lies in the complexity of Key Performance Indicators. KPIs cannot be evaluated in isolation, they must be interpreted in context. For example, emission levels are more critical when they occur near densely populated areas. Similarly, large variations in vehicle speed may be due to the presence of many intersections and do not necessarily indicate a major safety concern. Therefore, it is essential to analyze the KPIs more deeply and visualize them using GIS software to gain clearer insights.

As mentioned earlier, graph theory proved to be a powerful tool in this study. It accurately identified the two main bottlenecks in the network. However, it struggles to capture more complex traffic dynamics, such as weaving sections, where traffic behavior is less predictable. Therefore, also a dynamic OD-matrix should be used to see how the local agent will learn when traffic demands are changing even more throughout one episode. This again highlights the need for well calibrated OD-matrix preferable ones that respond to outdoor measurements from detectors.

Furthermore, more analysis is needed to understand the impact of the signal reward function assigned by the global agent, particularly how it influences the overall learning process and coordination between intersections. Coordination in timing between downstream and upstream intersections would greatly enhance the performance of local agents by distributing flows more effectively and prioritizing routes that are free from traffic jams.

The new optimal configuration reached under this centralized reward system may differ significantly from what would be achieved using local agents that do not communicate and focus solely on pursuing their own goals. Exploring how different reward structures affect the system's behavior is essential for fine-tuning performance and ensuring fairness and efficiency in complex traffic scenarios.

The most important focus for future research should be scalability. Applying this approach to larger, real-world urban networks remains a key long-term objective. To make this feasible, further investigation into hierarchical learning frameworks or modular training strategies will be essential. This thesis demonstrates that the use of a double-layered neural network holds significant potential—a potential that is likely to increase as the network size grows. This is linked closely to the further research needed to make digital twins that represent reality and use the input from detectors in the streets to feed the OD-matrix in the simulation.

This research introduces two significant new points: first, the extended formulas for utilizing alternative paths to mitigate the importance of a given link. Most of the existing literature focuses solely on the repercussions of a link being cut and whether certain regions become isolated as a result. The impact of alternative routes has not been sufficiently studied and therefore the parameter of 2/3 is used, but it requires further refinement in future work.

The second key takeaway from this study is the potential of reinforcement learning as an attractive tool for optimizing traffic lights, a concept that is already supported by many researchers (see Chapter 3). However, many DQN structures fail to address broader societal goals. The new approach, which emphasizes a visual framework, offers the most valuable insight gained from this research. The dashboard facilitates an interactive loop between users and the optimization process, allowing users to explore and evaluate how the preferred system performs.

References

[1] Juan Lu et al. "Expansion of city scale, traffic modes, traffic congestion, and air pollution". In: *Cities* 108 (2021), p. 102974. ISSN: 0264-2751. DOI: https://doi.org/10.1016/j.cities.2020.102974. URL: https://www.sciencedirect.com/science/article/pii/S0264275120313226.

[2] Julia Dratva et al. "Impact of road traffic noise annoyance on health-related quality of life: results from a population-based study". In: *Quality of Life Research* 19.1 (2010), pp. 37–46. ISSN: 1573-2649. DOI: 10.1007/s11136-009-9571-2. URL: https://doi.org/10.1007/s11136-009-9571-2.

[3] Donggyun Ku et al. "Review of European Low Emission Zone Policy". In: *Chemical Engineering Transactions* 78 (Apr. 2020), pp. 78, 241–246. DOI: 10.3303/CET2078041.

[4] Wei Hu et al. "A swarm intelligent method for traffic light scheduling: application to real urban traffic networks". In: *Applied Intelligence* 44.1 (Jan. 2016). Published online: 05 August 2015, pp. 208–231. DOI: 10.1007/s10489-015-0701-y. URL: https://doi.org/10.1007/s10489-015-0701-y.

[5] Shima Damadam et al. "An Intelligent IoT Based Traffic Light Management System: Deep Reinforcement Learning". In: *Smart Cities* 5.4 (2022). Submitted: 15 August 2022; Revised: 20 September 2022; Accepted: 22 September 2022; Published: 27 September 2022, pp. 1293–1311. DOI: 10.3390/smartcities5040066. URL: https://doi.org/10.3390/smartcities5040066.

[6] Julián Hurtado-Gómez et al. "Traffic Signal Control System Based on Intelligent Transportation System and Reinforcement Learning". In: *Electronics* 10.19 (2021). Submitted: 30 August 2021; Revised: 24 September 2021; Accepted: 24 September 2021; Published: 28 September 2021, p. 2363. DOI: 10.3390/electronics10192363. URL: https://doi.org/10.3390/electronics10192363.

[7] Chenghao Li et al. "Fairness Control of Traffic Light via Deep Reinforcement Learning". In: *2020 IEEE 16th International Conference on Automation Science and Engineering (CASE)*. 2020, pp. 652–658. DOI: 10.1109/CASE48305.2020.9216899.

[8] Chrysostomos Mylonas, Evangelos Mitsakis, and Konstantinos Kepaptsoglou. "Criticality analysis in road networks with graph-theoretic measures, traffic assignment, and simulation". In: *Physica A: Statistical Mechanics and its Applications* 629 (2023), p. 129197. ISSN: 0378-4371. DOI: https://doi.org/10.1016/j.physa.2023.129197. URL: https://www.sciencedirect.com/science/article/pii/S0378437123007525.

[9] Satu Elisa Schaeffer. "Graph clustering". In: *Computer Science Review* 1.1 (2007), pp. 27–64. ISSN: 1574-0137. DOI: https://doi.org/10.1016/j.cosrev.2007.05.001. URL: https://www.sciencedirect.com/science/article/pii/S1574013707000020.

[10] Urška Demšar, Olga Špatenková, and Kirsi Virrantaus. "Identifying Critical Locations in a Spatial Network with Graph Theory". In: *Transactions in GIS* 12 (Feb. 2008), pp. 61–82. DOI: 10.1111/j.1467-9671.2008.01086.x.

[11] Junchen Jin and Xiaoliang Ma. "Hierarchical multi-agent control of traffic lights based on collective learning". In: *Engineering Applications of Artificial Intelligence* 68 (2018), pp. 236–248. ISSN: 0952-1976. DOI: https://doi.org/10.1016/j.engappai.2017.10.013. URL: https://www.sciencedirect.com/science/article/pii/S0952197617302658.

[12] Dimitrius F. Borges et al. "Traffic Light Control Using Hierarchical Reinforcement Learning and Options Framework". In: *IEEE Access* 9 (2021), pp. 99155–99165. DOI: 10.1109/ACCESS.2021.3096666.

[13] Zhiyi Li et al. "A Hierarchical Framework for Intelligent Traffic Management in Smart Cities". In: *IEEE Transactions on Smart Grid* 10.1 (2019), pp. 691–701. DOI: 10.1109/TSG.2017.2750542.

[14] M.G.H. Bell I. Kaparias. *Key Performance Indicators for traffic management and Intelligent Transport Systems*. June 2011.

[15] A. Tsakarestos N. Eden I. Kaparias. *Using Key Performance Indicators for Multi-Criteria Traffic Management Strategic Decisions*. June 2013.

[16] Matthias Sweet. “Does Traffic Congestion Slow the Economy?” In: *Journal of Planning Literature* 26 (4 Oct. 2011). doi: 10.1177/0885412211409754, pp. 391–404. ISSN: 0885-4122. DOI: 10.1177/0885412211409754. URL: https://doi.org/10.1177/0885412211409754.

[17] S. van Duren L. Rijsman. *Fietsberaadnotitie: Meer ruimte voor fietsers en voetgangers*. Mar. 2021.

[18] Theodore J. Schultz. “Synthesis of social surveys on noise annoyance”. In: *The Journal of the Acoustical Society of America* 64.2 (Aug. 1978). Available to Purchase, pp. 377–405. DOI: 10.1121/1.382013. URL: https://doi.org/10.1121/1.382013.

[19] Paulo Henrique Trombetta Zannin, Fabiano Belisário Diniz, and Wiliam Alves Barbosa. “Environmental noise pollution in the city of Curitiba, Brazil”. In: *Applied Acoustics* 63.4 (2002), pp. 351–358. ISSN: 0003-682X. DOI: https://doi.org/10.1016/S0003-682X(01)00052-4. URL: https://www.sciencedirect.com/science/article/pii/S0003682X01000524.

[20] Christian Maschke, Karl Hecht, and Hans□Ulrich Balzer. “Preventative medical limits for chronic traffic noise exposure”. In: *The Journal of the Acoustical Society of America* 105.2 (Feb. 1999), pp. 1374–1374. ISSN: 0001-4966. DOI: 10.1121/1.426504. URL: https://doi.org/10.1121/1.426504.

[21] Weijun Yang et al. “Evaluation of urban traffic noise pollution based on noise maps”. In: *Transportation Research Part D: Transport and Environment* 87 (2020), p. 102516. ISSN: 1361-9209. DOI: https://doi.org/10.1016/j.trd.2020.102516. URL: https://www.sciencedirect.com/science/article/pii/S1361920920307033.

[22] Panos G. Michalopoulos and George Stephanopoulos. “Oversaturated signal systems with queue length constraints—I: Single intersection”. In: *Transportation Research* 11 (6 Dec. 1977), pp. 413–421. ISSN: 0041-1647. DOI: 10.1016/0041-1647(77)90006-5.

[23] Henry X. Liu et al. “Real-time queue length estimation for congested signalized intersections”. In: *Transportation Research Part C: Emerging Technologies* 17 (4 Aug. 2009), pp. 412–427. ISSN: 0968-090X. DOI: 10.1016/J.TRC.2009.02.003.

[24] Hameed A. Mohammed, Masoud Ghodrat Abadi, and David S. Hurwitz. “Red-light running violation during car following at high-speed signalized intersections”. In: *Transportation Engineering* 8 (2022), p. 100110. ISSN: 2666-691X. DOI: https://doi.org/10.1016/j.treng.2022.100110. URL: https://www.sciencedirect.com/science/article/pii/S2666691X22000082.

[25] H. van Gurp. *De verkeerskundige staat van de stad. Een onderzoek naar de verandering van de presentatie van het wegennet tussen 2011 en 2013 in de Gemeente Enschede*. Aug. 2014. URL: http://essay.utwente.nl/66368/.

[26] H.J. Harms. *Bicycle-friendly traffic light regime - evaluation*. Mar. 2008.

[27] Q. He Z. Zhang. “Performance measure for reliable travel time of emergency vehicles”. In: *Transportation Research Part C: Emerging Technologies* 65 (2016), pp. 97–110. ISSN: 0968-090X. DOI: https://doi.org/10.1016/j.trc.2016.01.015.

[28] Roderick B. Diaz and Donald C. Schneck. “Bus Rapid Transit Technologies in the Americas: An Overview”. In: *Transportation Research Record* 1731.1 (2000), pp. 3–9. DOI: 10.3141/1731-01. eprint: https://doi.org/10.3141/1731-01. URL: https://doi.org/10.3141/1731-01.

[29] Gemeente Rotterdam. *Fietskoers 2025, nog meer Rotterdammers op de fiets*. July 2019. URL: https://persberichtenrotterdam.nl/persberichten/fietskoers-2025-nog-meer-rotterdammers-op-de-fiets/.

[30] Gemeente Eindhoven. *Snelheidsverlaging binnen de bebouwde kom*. 2022. URL: https://www.eindhoven.nl/stad-en-wonen/verkeer/verkeersveiligheid/snelheidsverlaging-binnen-de-bebouwde-kom.

[31] Gemeente Almelo. *Mobiliteitsvisie Almelo: 15 Uitgangspunten*. Apr. 2025. URL: https://beleidsradar.nl/documenten/mobiliteitsvisie-almelo-15-uitgangspunten-a8dcf426-453a-4420-8fb8-49b5a6a95938.

[32] Lou Janssen. "Unlocking Shared Mobility: A Transition Governance Perspective on the Role of Local Governments in the Netherlands". Master's thesis. Eindhoven: Eindhoven University of Technology, Sept. 2023.

[33] David Levinson and. "Equity Effects of Road Pricing: A Review". In: *Transport Reviews* 30.1 (2010), pp. 33–57. DOI: 10.1080/01441640903189304. URL: https://doi.org/10.1080/01441640903189304.

[34] brabantgaatvoornul. *Brabant gaat voor NUL verkeersslachtoffers!* Date not specified. URL: https://brabantgaatvoornul.nl/default.aspx.

[35] Tibor Rongen et al. "An analysis of the mobility hub concept in the Netherlands: Historical lessons for its implementation". In: *Journal of Transport Geography* 104 (2022), p. 103419. ISSN: 0966-6923. DOI: https://doi.org/10.1016/j.jtrangeo.2022.103419. URL: https://www.sciencedirect.com/science/article/pii/S0966692322001429.

[36] Gemeente Den haag. *Buurthubs*. Date not specified. URL: https://bereikbarestad.denhaag.nl/buurthub/.

[37] Gemeente Utrecht. *Jouw straat en onze stad gezond, aantrekkelijk en bereikbaar voor iedereen*. July 2021. URL: https://omgevingsvisie.utrecht.nl/fileadmin/uploads/documenten/zz-omgevingsvisie/thematisch-beleid/verkeer-mobiliteit/2021-07-mobiliteitsplan-2040.pdf.

[38] Steve Kearns et al. *STRIA Roadmap Network and Traffic Management Systems Expert Group Rapporteurs*. Nov. 2016.

[39] G Sarlas and A Kouvelas. "Analysis of Urban Traffic Network Vulnerability and Classification of Signalized Intersections". In: *2019 6th International Conference on Models and Technologies for Intelligent Transportation Systems (MT-ITS)*. 2019, pp. 1–6. DOI: 10.1109/MTITS.2019.8883363.

[40] Ting Lu, Dan Zhao, and Yanhong Yin. "An Intersection Importance Estimation Method Based on the Road Network Topology". In: *The Open Civil Engineering Journal* 12 (2018), pp. 239–249. ISSN: 1874-1495. DOI: https://doi.org/10.2174/1874149501812010239. URL: https://www.sciencedirect.com/science/article/pii/S1874149518000038.

[41] Mohammad Reza Valipour Malakshah and Zahra Amini. *Towards an Operationally Resilient Urban Traffic Network: An Intersection Importance Assessment Framework Through MCDM Methods*. Available at SSRN: https://ssrn.com/abstract=5064047. Aug. 2024. DOI: 10.2139/ssrn.5064047. URL: https://ssrn.com/abstract=5064047.

[42] Randolph William Hall. "Traveler route choice: Travel time implications of improved information and adaptive decisions". In: *Transportation Research Part A: General* 17.3 (1983), pp. 201–214. ISSN: 0191-2607. DOI: https://doi.org/10.1016/0191-2607(83)90042-0. URL: https://www.sciencedirect.com/science/article/pii/0191260783900420.

[43] David C. Adam et al. "Clustering and superspreading potential of SARS-CoV-2 infections in Hong Kong". In: *Nature Medicine* 26.11 (Nov. 2020). Received 19 May 2020; Accepted 02 September 2020; Published 17 September 2020, pp. 1714–1719. DOI: 10.1038/s41591-020-1092-0. URL: https://doi.org/10.1038/s41591-020-1092-0.

[44] Ante Pusic. *Graph Clustering Algorithms: Usage and Comparison*. https://memgraph.com/blog/graph-clustering-algorithms-usage-comparison. 9 min read, May 26, 2023. 2023.

[45] NetworkX Developers. *networkx.algorithms.cluster.clustering — NetworkX Documentation*. Accessed: 2025-04-01. 2024. URL: https://networkx.org/documentation/stable/reference/algorithms/generated/networkx.algorithms.cluster.clustering.html.

[46] E.A. Mueller. "Aspects of the history of traffic signals". In: *IEEE Transactions on Vehicular Technology* 19.1 (1970), pp. 6–17. DOI: 10.1109/T-VT.1970.23426.

[47] Jakub Palša et al. "Smart Cities and the Importance of Smart Traffic Lights". In: *2019 17th International Conference on Emerging eLearning Technologies and Applications (ICETA)*. 2019, pp. 587–592. DOI: 10.1109/ICETA48886.2019.9040086.

[48] Hua Wei et al. "Recent Advances in Reinforcement Learning for Traffic Signal Control: A Survey of Models and Evaluation". In: *SIGKDD Explor. Newsl.* 22.2 (Jan. 2021), pp. 12–18. ISSN: 1931-0145. DOI: 10.1145/3447556.3447565. URL: https://doi.org/10.1145/3447556.3447565.

[49] Patrick Mannion, Jim Duggan, and Enda Howley. "An Experimental Review of Reinforcement Learning Algorithms for Adaptive Traffic Signal Control". In: *Autonomic Road Transport Support Systems*. Ed. by Thomas Leo McCluskey et al. Cham: Springer International Publishing, 2016, pp. 47–66.

[50] Monireh Abdoos and Ana L.C. Bazzan. "Hierarchical traffic signal optimization using reinforcement learning and traffic prediction with long-short term memory". In: *Expert Systems with Applications* 171 (2021), p. 114580. ISSN: 0957-4174. DOI: https://doi.org/10.1016/j.eswa.2021.114580. URL: https://www.sciencedirect.com/science/article/pii/S095741742100021X.

[51] Omid Nayeri et al. "Traffic signal optimization using hierarchical reinforcement learning: incorporating pedestrian dynamics and flashing light mode". In: *Transportation Planning and Technology* 0.0 (2025), pp. 1–30. DOI: 10.1080/03081060.2025.2457030. URL: https://doi.org/10.1080/03081060.2025.2457030.

[52] Zach Bobbitt. *How to Interpret Cohen's d (With Examples)*. Last updated on August 31, 2021. 2021. URL: https://www.statology.org/interpret-cohens-d/ (visited on 07/13/2025).

[53] Nettie Bakker. *Gemiddelde autorit is 19 kilometer en duurt een kwartier*. Published on December 27, 2019, in Verkeerskunde. 2019. URL: https://www.verkeerskunde.nl/artikel/gemiddelde-autorit-is-19-kilometer-en-duurt-een-kwartier (visited on 07/13/2025).

[54] Gregory Kovalchuk. *A Beginner's Guide to Q-Learning: Understanding with a Simple Gridworld Example*. Published on October 24, 2024. 2024. URL: https://medium.com/@goldengrisha/a-beginners-guide-to-q-learning-understanding-with-a-simple-gridworld-example-2b6736e7e2c9 (visited on 07/13/2025).

[55] Mahdi Amouzadi, Mobolaji Olawumi Orisatoki, and Arash M. Dizqah. "Capacity Analysis of Intersections When CAVs Are Crossing in a Collaborative and Lane-Free Order". In: *Future Transportation* 2.3 (2022), pp. 698–710. ISSN: 2673-7590. URL: https://www.mdpi.com/2673-7590/2/3/39.

[56] Daehyon Kim and Yohan Chang and. "Neural Network-based O-D Matrix Estimation from Link Traffic Counts". In: *International Journal of Urban Sciences* 12.2 (2008), pp. 146–157. DOI: 10.1080/12265934.2008.9693637. eprint: https://doi.org/10.1080/12265934.2008.9693637. URL: https://doi.org/10.1080/12265934.2008.9693637.

[57] Feng Li et al. "Dynamic traffic noise simulation at signal-controlled intersections based on cellular automata model". In: *Applied Acoustics* 224 (2024), p. 110128. ISSN: 0003-682X. DOI: https://doi.org/10.1016/j.apacoust.2024.110128. URL: https://www.sciencedirect.com/science/article/pii/S0003682X24002792.

[58] Natalia Sobrino, Andrés Monzón, and Sara Hernandez. "Reduced Carbon and Energy Footprint in Highway Operations: The Highway Energy Assessment (HERA) Methodology". In: *Netw Spat Econ* 16 (Apr. 2014), pp. 1–20. DOI: 10.1007/s11067-014-9225-y.

A
OD-callibration

As was discussed in the literature, updating the OD-matrix is essential to improve microsimulation accuracy. In addition, this also allows the simulation to better adapt when it measures changes at the beginning of the network. Then it will be able to predict the consequences downstream. This allows the network to respond very well to sudden but large changes. [56] as mentioned, using a relatively simple neural network with Backpropagation Momentum is already quite good at predicting these original OD-matrix. For both networks, the counters will be in the detector 1 meter in front of the traffic light. Firstly, it was tried not to implement the network layout in the neural network to make it as simple as possible and to see how it performs. It was found that for relatively small networks this did work, but when the network became more complex, instabilities were found.

To simplify the training, it is assumed that an initial OD-matrix can be estimated (in practice a static model is already a good start). Another simplification made is that the input and output should only be a static OD-matrix. In practice a time-dependent OD matrix would be more insightful for the real-time traffic manager. When the network is considered, relation between lengths and speed could be the key to add help by changing the OD-matrix from a static one into a dynamic one. This is a good addition for future research into this specific topic. For network 1 a random real world OD-matrix is used (step 0), since this network is randomly created and is not based on reality. For the string in Almelo Omnitrans was used to estimate the real-world OD-matrix (Step 0). Normally the static model would be used to estimate Step 2, here because real-world detector data was missing it is assumed that this OD matrix is in fact correct. Omnitrans uses a gravity model that takes into account the capacity constraint for each link and distributes traffic according to the gravity model.

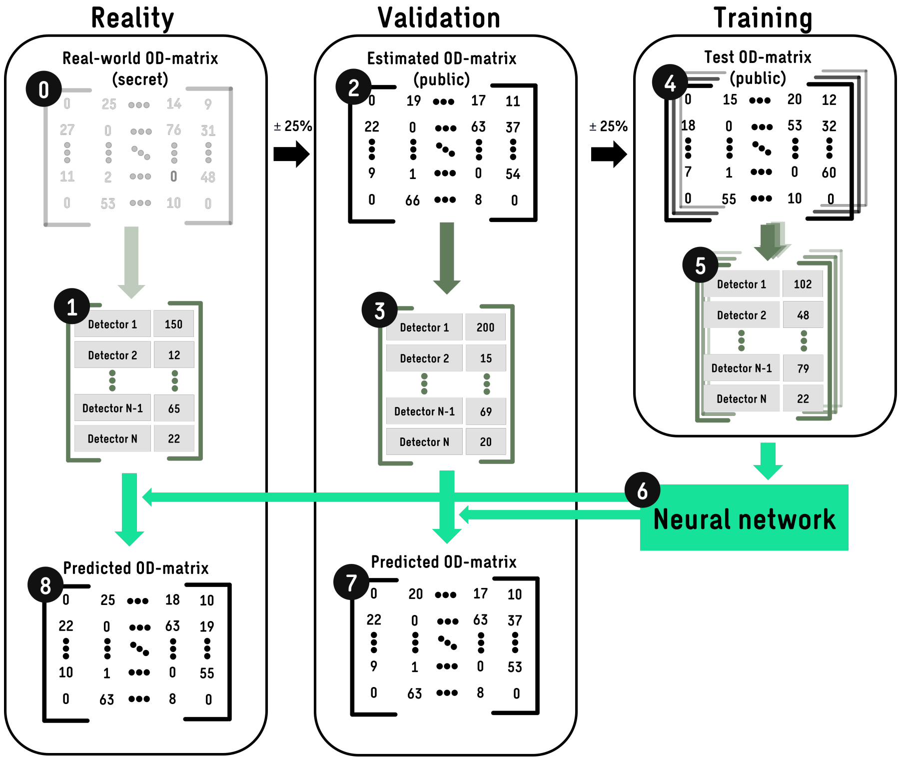


Figure A.1: OD calibration framework

As shown in the figure above (A.1), the real world OD-matrix is the "true" matrix. This matrix is unknown and this process will try to estimate this matrix, which is therefore also the last step in the framework. Step 1 shows the detector data as measured from the loop detectors in the streets. Step 2 shows the matrix that can be calculated using a static model, and here it is assumed that the OD-relation can vary between 0-25% higher or lower values. This model can be run to find the detector data in the Simulation software. This detector data should be taken at the same location as in the real-world scenario to get a good representation.

To train the neural network, the step from 0 to step 2 is again repeated to find the training dataset that has significant variations to capture a shift in demand. The amount of matrices can be varied to have a larger or smaller training data set (a maximum of 25% was assumed here). Important to note here is that the more complicated the model is, the longer the simulations will take, and the more simulation will be required. Step 5 shows the detector data for all the different simulations that were performed. The neural network will take as input the detector data and as output the test OD-matrix. The more data available, the better the prediction will be. At a certain point, overfitting can become a problem, which would mean that the model is not able to adapt to new situations. When the neural network is trained, the loop detector data in step 3 can be used to predict a new matrix. The accuracy of the neural network is then calculated as the difference between Steps 2 and 7. This validation step will show how well the neural network performs. Using the same neural network in Step 1 will estimate the real-world OD-matrix which is unknown. This predicted OD matrix should be closer to the estimate than what the Omnitrans model would predict. In case of disruptions, the model should be able to adapt in case certain OD pairs see a lot more traffic. This will be a very powerful tool to use for the digital twin of reality to run possible scenarios in real time. To see how well the model really performs in the next paragraph, step 0 and step 8 will be compared for network 1 and 2 from only 5 test runs up to 200.

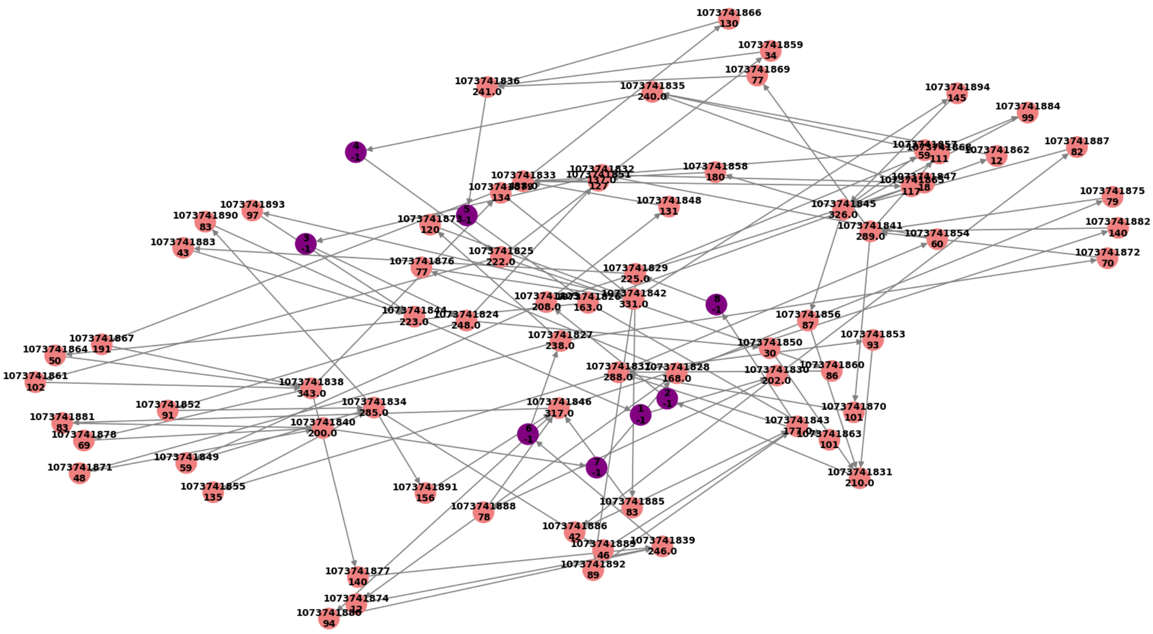


Figure A.2: Simple network graph representation

The neural network is a graph neural network with nodes; these nodes represent places (intersections and merge lanes) where traffic can diverge or converge. As can be seen in the above picture, the OD nodes are shown in purple, while all other nodes have their node number and traffic flow assigned to the node. At these nodes due to the use of these loop detectors, the amount of traffic is known. Before and after lane changes the traffic might be unknown; therefore, a smart technique of looking backward or forward is used to find the missing flow links (here shown with a question mark).

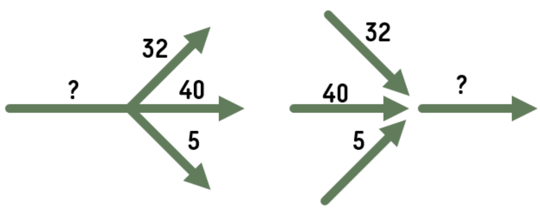


Figure A.3: Enter Caption

The model uses GATConv (Graph Attention Convolution) layers, which allow it to focus on the most important neighbors when updating each node's information. The special nodes are those where traffic originates or ends. The model uses these nodes to estimate how much traffic originated there and how the traffic traveled through the network. Additional heuristics were utilized to guide the model in the right direction. Negative values are penalized because the negative flow does not make sense.

For the start nodes, the sum of all traffic leaving should be equal to the flow measured on the first links departing from them. Similarly, for the end nodes, the amount of traffic arriving at the node should match the amount of traffic arriving via the incoming links. The model combines these heuristics with the objective of producing an OD (origin-destination) matrix that matches the original one in order to accurately predict OD flows from car loop data.

A.1. Performance measures of dector data to predict OD-matrix

To determine how many runs are required to obtain a reasonably accurate neural network, both simulations were executed 200 times. The neural network was trained using randomly selected subsets of data, starting with just 20 samples and increasing incrementally to the full dataset of 200. The selections were all randomly drawn for all increments so the set with 20 dataset and 40 could have been totally different. This approach allows us to observe how the model performance improves with the addition of more data. This can be the reason for the large differences in performance between certain models.

Each model was trained over 500 episodes and 30% of the simulation runs in each episode were used to train the model. Each episode used this 30% of the available data to estimate the values of the neural network. Using only a portion of the data is crucial to prevent overfitting. To evaluate the robustness of the model under extreme conditions, the following four changes were applied to selected pairs of OD (origin destination), this was performed separately:

- One OD-pair experienced a 50% decrease in traffic.
- Another OD-pair experienced a 75% decrease.
- A third OD-pair saw a 100% increase.
- A fourth OD-pair underwent a 200% increase.

All other OD-pairs remained unchanged. This setup allows us to isolate and observe the impact of individual OD pair changes on the model predictions. Figures A.4 and A.5 both show the performance of the model versus the size of the training. It is important to reiterate that batches with sizes such as 40 and 42, for example, can have completely different datasets since they were randomly drawn from the 200 simulations. This helps explain the large fluctuations, as the model sometimes trains with data that better fits the original matrix. Both models clearly show a downward trend, indicating that the model is indeed improving with more data.

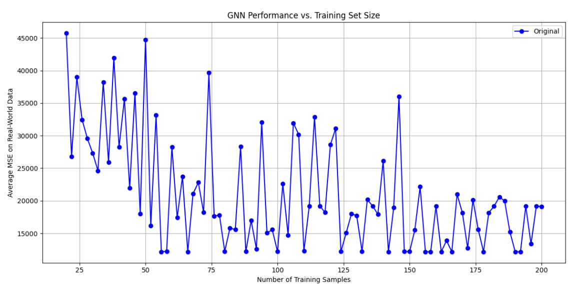


Figure A.4: Training improvements simple network

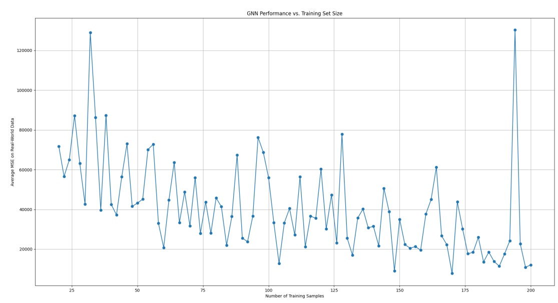


Figure A.5: Training improvements string Almelo

There are some important remarks to be made. The large peak in Figure A.5 at the end is likely due to the model occasionally starting in the wrong direction. Given the limited training time (as training 180 different models takes a long time), the number of training cycles was kept at 500, which could explain the large error. It can take a long time for the model to adapt if the model starts learning in the wrong direction because of certain simulation that was heavily skewed in a certain direction. The figure on the left also shows that the model reaches an equilibrium faster, since it has only 8 OD pairs, thus needing to predict 64 OD relations. In contrast, for the Almelo string, there are 324 OD pair combinations (18*18).

The errors are still squared, which explains the high numbers for the model error. The total number of cars on the simple network is 1,573, and the Almelo network sees 3,590 cars. This means that the total number of cars miscounted in the OD matrix is $\sqrt{17,500} = 132$, which leads to a percentage of just under 10%. For the Almelo network, this flattens out around 20,000, leading to an error of $\sqrt{20,000} = 141$ cars, which results in a percentage of just under 4%. These are pretty good results. When studying the impact increasing or decreasing the load on one od pair given the trained model, a better score was found also for the original scenario. For seeing the impact only 20 models were trained with more training cycles (see Figures A.8 and A.9). Both score significantly better also for the simple scenario and the error was reduced from 10% to 7 %.

The difference in both networks could be due to the neural network requiring more simulations to understand the use of alternative paths. Adding more information about alternative routes could help the network, making this an important aspect for future research on this topic.

The figure below shows the network performance in response to changes in demand. It is important to note that, for the simple network, the ranges may appear extensive, but the number of misrepresented cars is at most 126. Surprisingly, the simple network performs better in predicting unknown OD matrices in situations where traffic conditions have changed from their original state. Notably, the simple network is slightly more accurate at predicting scenarios with decreased traffic than those with increased traffic. Further research will be conducted on the performance of the other network.

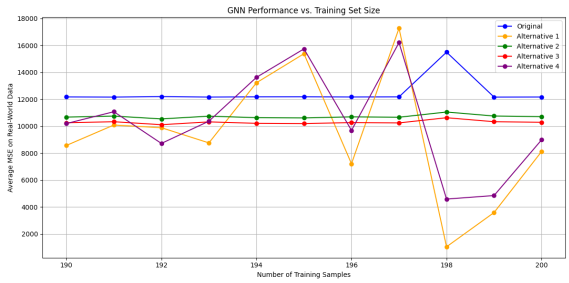


Figure A.6: Alternatives scoring simple network

Figure A.7: Alternatives simple network shown in map

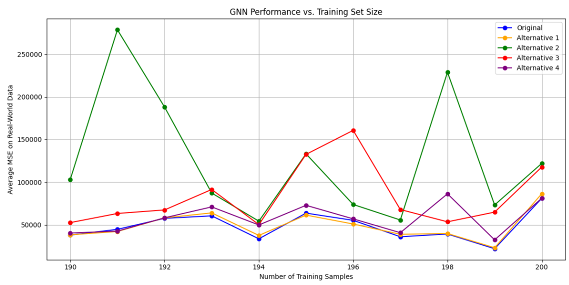


Figure A.8: Alternatives scoring simple network

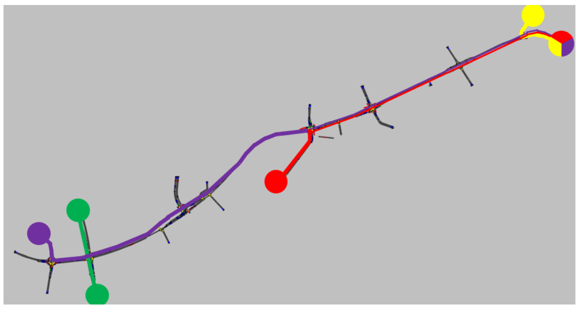


Figure A.9: Alternatives Almelo shown in map

B

Visvap software

VisVap is the flow diagram-based way of writing a program to make a level 2/3 traffic lights. Visvap is a tool specificity designed to work with the PTV Vissim micro simulation software. This software was used for the traffic lights that exist in the Almelo network. First, for all traffic lights the main signal groups were determined. Secondly, the main structure was developed so that these traffic groups can follow each other in a set order. Third, the posibility to extend green times and to skip stages was implemented. For example, the following structure is used stage 1 -> stage 2 -> stage 3 in the case that stage 2 sees no traffic the following structure is used 1->3.

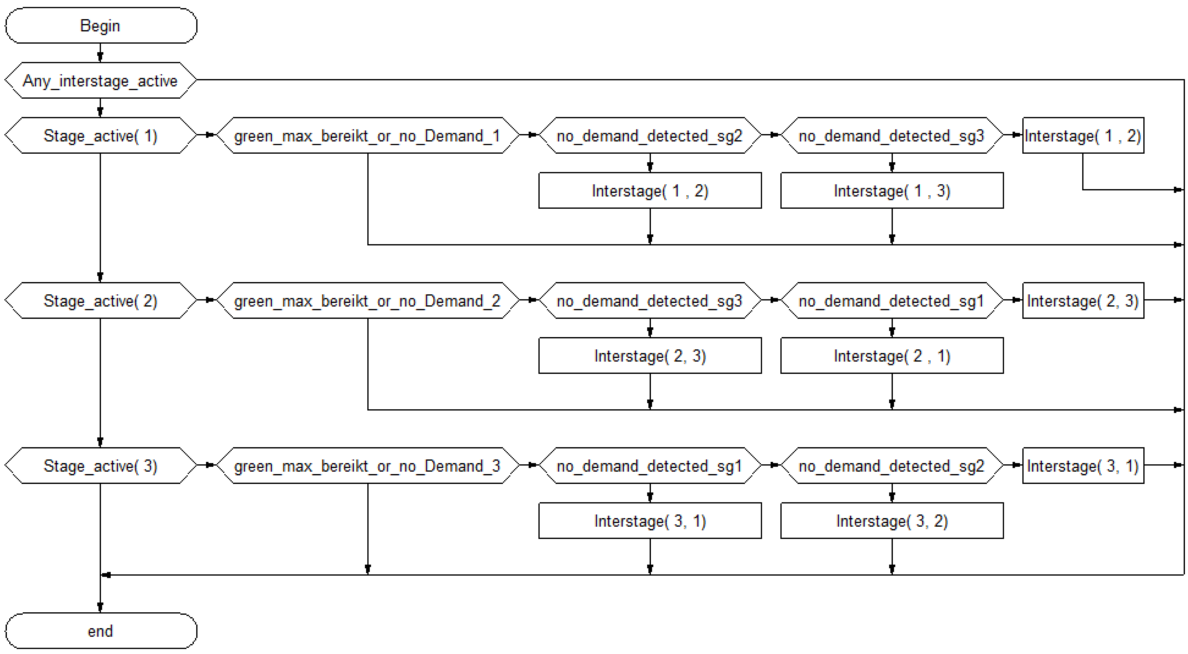


Figure B.1: VisVap simple structure

There is only one slight problem with this reasoning in relation to safety. When going from stage 1 -> stage 3 takes relatively long since there is a pedestrian that needs an intergreen of 17 seconds, the visvap does not consider this since it only looks at the intergreen in between the active and next stage. When going from stage 1 towards stage 2 is short, say 8 seconds (includes 5 seconds of minimum green time) and going from stage 2 towards stage 3 takes only 7 seconds (incl. min green time). When going from 1->2->3 because of these short intergreen times, the intergreen between 1 and 3 is violated since it is only 15 (7+8), while the lower limit has to be 17 to be considered safe. Therefore, the program was extended to incorporate this by starting timers and checking if the timer has reached the minimum time between 1 and 3. This complicates things, but is necessary to comply with safety regulations. This flow diagram is created for all six intersections using Python to generalize the process. In addition, the safety regulation check is performed by Python to ensure that no human oversight can take place.

C
KPIs

C.1. Different KPIs

Translating raw measurements into meaningful scores remains a challenge. To evaluate performance across different scenarios, three distinct methods have been developed. Each method uses a standardized scoring scale from 0 to 10, where higher scores consistently indicate better performance. For metrics where "lower is better," the scale is reversed during processing to maintain this interpretation—ensuring that, after adjustment, a higher score still reflects better outcomes. The following sections describe each method in detail, including how they handle different performance criteria and apply the scoring adjustments accordingly.

Method 1: Score Based on Relative Performance This method categorizes performance on a scale from 0 to 10 based on relative comparisons. The scoring is defined as follows:

- 0: Represents a performance that is twice as bad as a baseline.
- 2.5: Indicates a performance that is 50% worse than the baseline.
- 5: This score suggests that nothing significant happened, meaning performance remained at a baseline level.
- 7.5: Reflects a performance that is 50% better than the baseline.
- 10: Denotes a performance that is twice as good as the baseline.

Method 2: Goal Achievement Scoring This method focuses on achieving defined performance goals and uses a similar scale 0 to 10. The scoring is established as follows:

- 0: Signifies a performance score of 0.
- 7: Represents the point at which a goal is reached
- 7+: Any score above 7 indicates that the performance has surpassed the goal, with linear scaling reflecting the degree of surpassing.

Method 3: Scaled Performance Metrics This method also adopts a scoring system from 0 to 10, but it uses a different set of criteria for performance measurement:

- 0: Indicates a performance that is twice as bad relative to a reference.
- 2.5: Represents a performance that is the worse performing and will be used as the baseline as the minimum in the base scenario.
- 5: Suggests that nothing significant happened, akin to a neutral performance level.
- 7.5: Represents a performance that is the best performing and will be used as the baseline as the maximum in the base scenario.
- 10: Demonstrates a performance that is four times better than the baseline.

All three methods have strengths and weaknesses. One significant drawback of the first method is that each intersection starts with a score of 5, making it difficult to distinguish which intersections operate at capacity and which are not. However, this method does allow for an understanding of the relative increases and decreases compared to the original scenario.

The second method is very powerful when the focus is on reaching specific thresholds, such as achieving zero red light violations or reducing overall speed in the city center. The main challenge with this approach is that establishing these thresholds can be quite difficult and may require a comprehensive review of the literature on the topic.

The third option is the most user-friendly as it does not have the same shortcomings as the first two methods. It immediately highlights which intersections perform worse than others, since it ranks them from 2.5 - 7.5 in the original scenario. However, this could be misleading as it still requires expert judgment. Not all intersections are equal, and comparing all intersections within the network can oversimplify the complexities of the system, necessitating a more detailed inspection.

C.2. Methode calculating different KPIs

These are the methods used for each KPI in PTV Vissim.

C.2.1. Traffic performance – motor vehicles

- **First waiting:**
  - **Simulation:** Maximum queue length during a specific time step (currently used due to continuous simulation time; will be updated to ”first waiting” with RL and step-by-step simulation).
- **Average waiting time at intersection:**
  - **Simulation:** Average vehicle delay at the intersection.
- **Vehicle hours waiting at intersection:**
  - **Simulation:** Total vehicle delay at the intersection.
- **Standard deviation of travel times (different days):**
  - **Simulation:** Standard deviation of total travel time for cars across different simulation seeds.
- **Network throughput:**
  - **Simulation:** Total distance $\times$ time (car) / (vehicles arrived + vehicles active).
- **Congestion duration:**
  - **Simulation:** Average stopped delay.

C.2.2. Traffic performance – bicycles

- **First waiting:**
  - **Simulation:** Maximum queue length during a specific time step (updated to ”first waiting” with RL and step-by-step simulation).
- **Average waiting time at intersection:**
  - **Simulation:** Average cyclist delay at the intersection.
- **Standard deviation of travel times (different days):**
  - **Simulation:** Standard deviation of total travel time for cyclists across different simulations.
- **Network throughput:**
  - **Simulation:** Total distance $\times$ time (cyclist) / (cyclists arrived + cyclists active).

C.2.3. Traffic performance – pedestrians

- **First waiting:**

  - **Simulation:** Maximum queue length during a specific time step (updated to ”first waiting” with RL and step-by-step simulation).

- **Average waiting time at intersection:**
  - **Simulation:** Average pedestrian delay at the intersection.
- **Standard deviation of travel times (different days):**
  - **Simulation:** Standard deviation of total travel time for pedestrians across different simulations.
- **Network throughput:**
  - **Simulation:** Total distance $\times$ time (pedestrian) / (pedestrians arrived + pedestrians active).

C.2.4. Air Quality

- **Carbon monoxide (CO):**
- **Nitrogen dioxide (NO_2):**
- **Particulate matter ($PM_{2.5}$):**
- **Volatile organic Compounds (VOCs):**
- **Carbon dioxide (CO_2):** (See appendix D for more information)

C.2.5. Safety of Roads

- **Queue Length:**
  - **Simulation:** Speed is less than 50% of free-flow speed.
- **Vehicle Kilometres:**
- **Light Red Violations – Light Red:**
- **Light Red Violations – Dark Red:**
- **Light Red Violations – Early Start:**
- **Number of Accidents at Crossings:**
- **Intersection Obstructions:**

C.2.6. Public Transport

- **Number of Late Arrivals at Stops:**
- **Total Travel Time (Bus):**
- **Public Transit Reliability:**

C.2.7. Noise

- **dB Generated:** (See appendix D for more information)

C.2.8. Equity

- **Emergency Vehicle Priority:**
- **Public Transport Priority:**

D

Emissions calculated (noise and CO_2)

Emissions are all around us, and it is therefore essential to understand the potential improvements that can be made in this area. PTV Vissim does not offer an option to accurately calculate traffic emissions or noise. Unfortunately, the Enviver license was not functional, and as a result, an alternative method for estimating these emissions had to be used.

To assess the environmental impact of traffic across road segments, two models were implemented: one to estimate traffic noise and another to estimate fuel-based CO_2 emissions. These models were applied to traffic data containing segment identifiers, speeds, volumes, and time intervals.

D.1. Traffic noise calculation

In the first step, the data was cleaned, and traffic volumes were adjusted to match the intervals used for emission determination. According to the literature by F. Li in his paper on Dynamic Traffic Noise Simulation [57], the following formula is used to estimate noise levels generated at intersections:

$$L_0 = \begin{cases} 24.92 \log_{10}(v) + 27.96, & \text{Light vehicle} \\ 36.73 \log_{10}(v) + 16.44, & \text{Medium vehicle} \\ 29.71 \log_{10}(v) + 31.77, & \text{Heavy vehicle} \end{cases} \tag{D.1}$$

Using this formula and the traffic volume, a baseline estimation of vehicle emissions for each road segment can be made. To determine whether vehicles are accelerating or decelerating, the average speed of the following link is compared to the current link. If the upcoming link has a higher average speed, it is assumed that vehicles are accelerating. Conversely, if the speed is lower, vehicles are assumed to be decelerating.

The correction factors for accelerating and decelerating vehicles are also provided in the same paper:

$$L_{A(i,j)} = \begin{cases} 3.55, & \text{Light vehicle with } a > 0 \\ -2.80, & \text{Light vehicle with } a < 0 \\ 4.32, & \text{Medium vehicle with } a > 0 \\ 1.51, & \text{Medium vehicle with } a < 0 \\ 7.25, & \text{Heavy vehicle with } a > 0 \\ 3.20, & \text{Heavy vehicle with } a < 0 \end{cases} \tag{D.2}$$

By applying these correction factors based on traffic conditions, a more accurate representation of true noise levels can be calculated. The combined noise from light, medium, and heavy vehicle categories is computed using energy-based summation. Additionally, a minimum threshold of 45 dB(A) is applied to represent ambient background noise.

D.2. Fuel-based CO_2 emissions estimation

Sobrino et al. highlight that while direct measurement of $CO2$ is challenging, fuel consumption provides a practical proxy [58]. In her research on fuel consumption and highway operations, the relationship between vehicle speed and fuel consumption for various car types is analyzed. This relationship is visualized in Figure D.1, showing how fuel efficiency varies with speed for gasoline passenger cars.

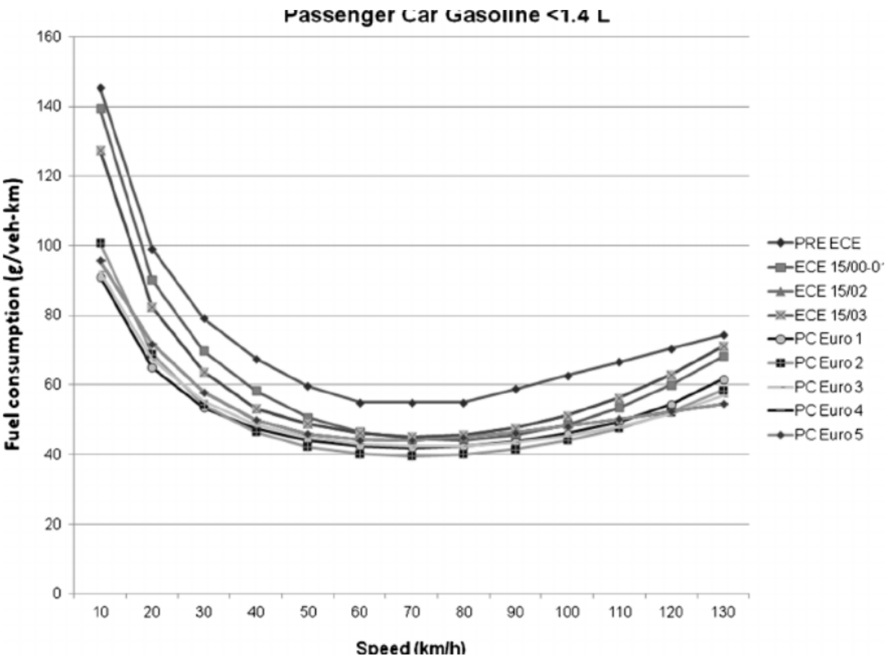


Figure D.1: Speed–fuel consumption curves for gasoline passenger cars by engine technology.

To estimate emissions, assumptions about the vehicle fleet composition and turnover were incorporated into the model:

- **Fleet turnover:** Each year, 5
- **Base year:** The model begins in 2025.
- **Vehicle lifespan:** Cars are assumed to retire after 12 years, with a maximum possible age of 40 years to account for anomalies.
- **Fuel type distribution in the Netherlands:**
  - Petrol vehicles: 65
  - Diesel vehicles: 29.6
  - Electric vehicles: 5.4

This information allows for an estimation of average fuel consumption and, by extension, average $CO2$ emissions per kilometer for the overall fleet.

To convert fuel consumption into $CO2$ emissions, an average emission factor of 2.3 kg $CO2$/liter of fuel is applied (based on the fuel-to-$CO2$ conversion shown in Figure D.2). This enables an approximation of tailpipe emissions across road segments.

Fuel type	CO_2 tailpipe emissions (kg/L)
Gasoline	2.29
E10 (10% ethanol + 90% gasoline)	2.21
E85 (85% ethanol + 15% gasoline)	1.61
Diesel	2.66
B5 (5% biodiesel + 95% diesel)	2.65
B20 (20% biodiesel + 80% diesel)	2.62

Figure D.2: Conversion from fuel type to tailpipe $CO2$ emissions.

While this model captures baseline emissions well, future work should incorporate the impact of acceleration and deceleration on fuel consumption. Such dynamic factors are particularly relevant near intersections, where emissions tend to spike and are more accurately modeled using microsimulation tools like PTV Vissim.

E
Graph theory generalisation

The graph theory formulas were initially applied to a relatively simple network as a proof of concept, and later extended to a more complex network in the city of Almelo. The simple network, illustrated in Figure E.1, serves as an effective demonstration of how graph-based analyses can be successfully implemented even in nonlinear, interconnected road systems. Unlike networks that follow a straightforward, linear structure, this example includes multiple intersections and varying road connections, thereby simulating real-world complexity on a smaller scale. This is crucial because it validates the applicability of graph theory beyond idealized or simplified cases, showing that these methods are not limited to abstract or academic scenarios but can be generalized to real-world transportation systems.

One notable anomaly observed in the simple network is the segment between the node with a score of 0.22 and the node with a score of 0.23. This road segment appears significantly redder, indicating a higher load or lower efficiency than would be expected based only on its spatial configuration. This discrepancy arises from the current shortest-path algorithm in use, which does not take into account the variation in speed limits across different road types. Specifically, the double-lane road connecting these nodes has a maximum speed of 70 km/h, whereas all other roads in the network have a speed limit of 50 km/h. As a result, the algorithm overestimates the cost (e.g., time or resistance) of using the higher-speed road. Incorporating speed as a weighting factor in the shortest-path calculations would refine the analysis, making the model more accurate by better reflecting actual travel conditions.

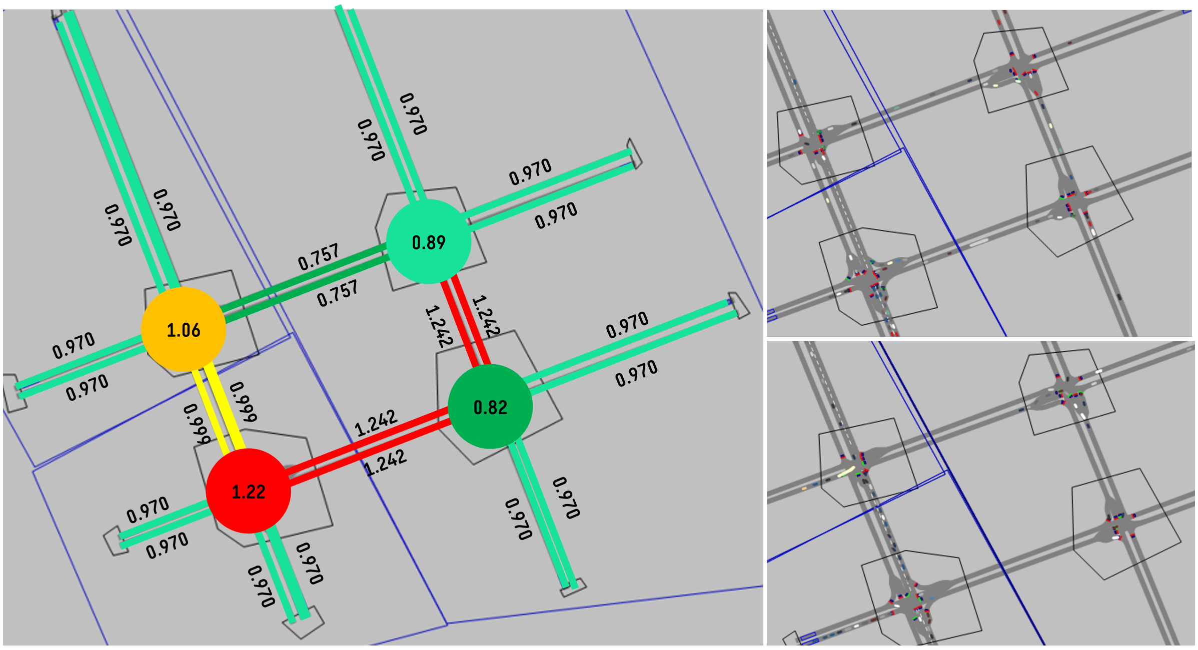


Figure E.1: Graph theory applied to simple network

In addition to graph theory, clustering techniques were also applied to the same network data. Interestingly, the results of the clustering closely mirrored the results obtained from the graph-theoretical analysis. This convergence of findings from two distinct methodologies highlights the robustness of the observed patterns. It suggests that the combination of graph theory and clustering provides a synergistic analytical framework. Although graph theory excels at quantifying connectivity, centrality, and flow efficiency, clustering helps identify communities or regions within the network that exhibit similar structural or functional characteristics. Together, these methods offer a more holistic view of the network, enabling more precise identification of bottlenecks, vulnerabilities, and optimization opportunities.

In summary, this multi-method approach demonstrates that graph theory is not only effective in simple, idealized networks but also generalizes well to more complex, realistic settings. When augmented with clustering techniques and potentially enhanced with additional real-world data inputs such as speed limits or traffic volume, this integrated analytical framework holds significant promise for advancing transportation planning, infrastructure optimization, and urban mobility management.