Zero-Shot Traffic Signal Control via Prophet Forecasting and Deep Q-Networks in SUMO Simulated Environments

Alcayde, Aidan Carl S., Helorentino, Rhonnmark L. and Manzon, Rod Vince B.

College of Computer Studies, Technological Institute of the Philippines Quezon City

CS 307 : Thesis 2

Ms. Janice Capule

September 2025

Chapter 1

Introduction

Background of the study

Traffic congestion remains a critical challenge across major urban centers worldwide, with cities experiencing exponential growth in vehicle volumes that far exceed infrastructure capacity. Urban highways and arterial roads face daily traffic loads that can exceed their design capacity by 30-40%, creating cascading effects of economic losses, environmental degradation, and reduced quality of life for millions of commuters. Metro Manila exemplifies this global urban mobility crisis, with major corridors like EDSA accommodating over 400,000 vehicles daily despite being designed for only 300,000 vehicles (Presidential Communications Office, 2024).

Initial attempts to manage this complexity relied on static, fixed-time signal schedules or, more recently, basic sensor-based reactive controllers. However, these systems are fundamentally limited, as they optimize for the immediate present and lack the mechanisms to anticipate and proactively manage incoming congestion. The literature points to a clear need for more intelligent, adaptive systems (Wang, Abdulhai, and Sanner, 2023). This has led to the rise of Reinforcement Learning (RL), a data-driven paradigm that allows an agent to learn optimal control policies through direct interaction with the environment. For this research, a Deep Q-Network (DQN) is selected as the core learning algorithm. The Review of Related Literature explores a range of advanced methods—from complex multi-agent policy-gradient approaches like DDPGAT and MADQN to sophisticated DQN variants like Double Dueling DQN (D3QN) and memory-based architectures like DRQN (Pálos & Huszák, n.d.; Rasheed et al., 2022;

Abdullah et al., 2023). While these state-of-the-art methods offer higher performance or specialized capabilities, they often come with significant implementation overhead. In contrast, a standard DQN offers a compelling and well-understood balance of performance and stability, providing a robust foundation for investigating the primary research objective of this thesis: creating a single, highly generalized agent.

However, the adoption of standard RL methodologies exposes a critical gap that hinders their scalability: non-transferability. As the literature on zero-shot learning emphasizes, conventional RL models are inherently brittle (Schmidt et al., 2024). They are typically trained to optimize a specific intersection, making them over-specialized. When applied to a new intersection with a different geometry or traffic profile, their performance collapses. This lack of transferability, often termed the reality gap or simulation-to-simulation gap, reveals the core problem this thesis addresses: no established pipeline exists for creating a traffic signal control agent that can be trained on a limited set of intersections and then generalize its learned knowledge to new, unseen traffic layouts without retraining (Müller & Sabatelli, 2023). To overcome this, modern research explores robust training methodologies. These include Domain Randomization (DR), where an agent is trained on a wide variety of randomized simulation parameters to ensure it generalizes to new, unseen environments, and Federated Learning, where multiple agents collaboratively train a shared model without centralizing data, improving learning speed and quality (Ye et al., 2021). Other advanced techniques involve creating robust policy ensembles using Distributional RL to better handle uncertainty and sensor failures (Shi et al., 2023). This thesis focuses specifically on Domain Randomization as the key technique to achieve zero-shot transfer.

To bridge this gap, this thesis adopts a zero-shot learning paradigm, a necessity for developing a truly scalable and viable traffic management solution. The objective is to create a single universal control policy that can be deployed “out-of-the-box” to any intersection. Achieving this requires careful consideration of forecasting tools, since predictive inputs shape the reinforcement learning process. Studies comparing ARIMA, LSTM, and Prophet show that ARIMA consistently achieves higher forecasting accuracy, with RMSE values as low as 5.5, while Prophet records higher error rates of around 8.02 (Uzel, 2023). Despite this performance gap, Prophet is strategically selected because its strengths lie in operational flexibility: it requires no dataset-specific tuning, handles missing and irregular data efficiently, and generates forecasts within milliseconds. Research further demonstrates that DQN-based control systems remain robust even with forecast errors up to 20% (Afandizadeh & Mirzahossein, 2024), placing Prophet’s accuracy well within acceptable limits for reinforcement learning integration. This trade-off prioritizes forecasting consistency, computational efficiency, and generalization over marginal accuracy gains, aligning directly with the zero-shot learning objective. Prophet’s ability to adapt seamlessly across diverse traffic environments ensures that the system transitions from reactive to proactive decision-making, enabling scalable deployment in heterogeneous urban intersections.

Finally, to develop and validate such a system, a powerful and flexible simulation environment is required. While various commercial platforms exist, the RRL justifies the selection of Simulation of Urban Mobility (SUMO) as the definitive open-source platform for modern traffic research. Its microscopic simulation engine, native Python and TraCI support for RL integration, and unrestricted access for implementing custom models and novel approaches like domain randomization make it the ideal sandbox. Unlike commercial alternatives, SUMO provides the necessary flexibility to build and rigorously test the generalizable, zero-shot agent that is at the heart of this research, addressing the historical lack of sophisticated, simulation-based evaluation in the local context.

Project Objectives

This study aims to implement a forecast-driven reinforcement learning pipeline that integrates time series forecasting with simulation-based adaptive traffic signal control. Prophet is used to generate short-term traffic demand forecasts from historical datasets, which serve as predictive inputs for a Deep Q-Network (DQN) agent trained within SUMO. During training, domain randomization is applied by varying intersection geometries, traffic flow distributions, and vehicle compositions to avoid overfitting and strengthen generalization. This combination of Prophet, SUMO, and DQN is selected for its balance of computational efficiency, interpretability, and stability, enabling the creation of signal control policies that can generalize to unseen intersections in SUMO simulations without requiring site-specific retraining.

1. To implement and validate a Prophet forecasting module that processes historical MMDA datasets into 24-hour traffic demand predictions, serving as direct state inputs for the DQN agent. Prophet is selected over ARIMA and LSTM due to its automatic seasonality detection, computational efficiency for real-time RL integration, and robustness to missing data patterns common in traffic datasets.
2. To establish a domain-randomized training environment where intersection geometries, traffic flow distributions, signal phase plans, and vehicle composition ratios are systematically varied. This approach prevents overfitting to single scenarios and enables

the DQN agent to learn robust, generalizable control policies capable of handling distributional shifts.

3. To design and train a DQN agent that maps forecast-augmented traffic states to optimal signal timing plan selection rather than second-by-second phase control. The DQN approach is preferred for its stability in discrete action spaces, safer alignment with predefined timing plans, and suitability for achieving zero-shot adaptability compared to more complex policy-gradient methods.
4. To evaluate the trained DQN policy on unseen intersections and traffic scenarios without any retraining or fine-tuning. The objective is to demonstrate that the learned policy can generalize effectively to new topologies and flow characteristics rather than memorizing specific training environments.
5. To evaluate the system using a two-tier framework that first measures traffic engineering metrics such as average waiting time, throughput, queue length, and fuel efficiency.
6. To evaluate the learned policy's zero-shot performance on simulated unseen intersections, comparing against fixed-timing and reactive baselines. using the ISO/IEC 25010 software quality characteristics of Functional Suitability, Performance Efficiency, Usability, and Reliability.

Significance of the Study

1. This study advances urban traffic management by developing a forecast-driven reinforcement learning system to optimize traffic signals along major intersections. The zero-shot capability represents a significant advancement, enabling rapid deployment of efficient traffic systems across urban environments without site-specific retraining. The

results will address current inefficiencies in congestion control and prove beneficial to the following:

2. Traffic management agencies can leverage the system's predictive capabilities to better plan and coordinate traffic signals, enabling a shift from reactive to anticipatory control and improving overall traffic flow. Commuters benefit from more accurate traffic forecasts that support better travel planning and reduced delays, potentially saving both time and fuel.
3. Urban planners and policymakers will find value in data-driven insights that highlight congestion hotspots, guiding infrastructure development and policy decisions. The system's analytical capabilities can identify patterns in traffic flow that inform long-term urban development strategies.
4. From an academic perspective, this research demonstrates the effective integration of machine learning, forecasting, and simulation techniques to solve real-world traffic challenges, laying a foundation for future innovations in smart traffic systems. The work provides a replicable framework that other researchers can adapt for different urban environments and traffic scenarios.
5. The research contributes to the advancement of Intelligent Transportation Systems (ITS) and smart city infrastructure. The integrated approach provides a blueprint for technology companies developing scalable traffic management solutions.
6. By optimizing traffic flow and reducing stop-and-go conditions, the system contributes to environmental sustainability through lower emissions and improved air quality. Studies show that optimized traffic signal control can reduce CO_2 emissions by 10-20% in urban areas.

7. The system's ability to reduce traffic congestion has direct economic implications for business productivity and transportation costs. Traffic congestion in major urban centers worldwide causes significant economic losses due to lost productivity and increased operational costs.
8. The adaptive nature of the traffic control system can be extended to prioritize emergency vehicles and improve overall road safety. The system's ability to adapt to unusual traffic patterns makes it valuable for managing traffic during emergencies or special events.
9. This research contributes to the broader movement toward data-driven governance and smart city initiatives. The system demonstrates how historical traffic data can be transformed into actionable intelligence for urban management.
10. The zero-shot learning capability addresses a critical gap in current traffic management research. This advancement enables rapid deployment across diverse urban environments without extensive retraining.
11. Vehicle operators and fleet managers can utilize the predictive insights to optimize routing decisions and reduce operational costs. The system's forecasting capabilities enable more efficient logistics planning and delivery scheduling.
12. Public transportation systems can benefit from improved traffic flow coordination, leading to more reliable transit schedules. Better signal timing reduces delays for mass transit vehicles across various urban environments.
13. Environmental agencies can leverage the system's emission reduction capabilities to support air quality improvement initiatives. The reduced idle time at intersections directly contributes to cleaner urban air.

14. Local governments can use the system's data analytics to support evidence-based decision making for infrastructure investments. The comprehensive traffic analysis provides crucial input for budget allocation and development priorities.
15. Tourism and business sectors benefit from improved traffic flow that enhances the overall urban experience. Reduced congestion makes cities more attractive for business investment and tourism development.
16. Finally, by optimizing traffic flow and reducing stop-and-go conditions, the system offers economic benefits by improving fuel efficiency and commuter productivity across urban areas globally.

Scope and Delimitations

Scope

This research develops an intelligent traffic management system that integrates predictive analytics with reinforcement learning to optimize urban intersection performance. The following are the scope of this study:

- In forecasting, the study focuses on short-term traffic forecasting (up to 24 hours ahead) using data from selected intersections along using Prophet.
- Historical traffic volume data will be obtained primarily from publicly available MMDA records or through Freedom of Information (FOI) requests.
- SUMO will be used to create digital models of selected intersections for training, including traffic flow patterns, lane configurations, and signal phases. In applying the final product or model, intersections unseen by the model will also be modeled and simulated in SUMO.

- The Deep Q-Learning (DQN) based reinforcement learning agent will be trained in SUMO using forecasted traffic demand at selected intersections and then tested for zero-shot generalization to unseen intersection layouts and traffic profiles that have similar qualities to the training set
- Model evaluation will include both forecasting accuracy metrics (e.g., MAE, RMSE) and simulation-based performance metrics aligned with the ISO/IEC 25010 software quality characteristics, specifically Functional Suitability, Performance Efficiency, Usability, and Reliability. (International Organization for Standardization, 2023)

Delimitations

This research develops an intelligent traffic management system that integrates predictive analytics with reinforcement learning to optimize urban intersection performance. The following are the limitations of this study:

- The research will not involve live deployment or direct control of real-world traffic signals; all testing will be conducted within a simulated SUMO environment.
- Real-time, continuous data collection will not be implemented; only historical datasets will be used for model training and evaluation.
- External, unstructured factors such as road accidents, weather conditions, or construction activities will be excluded unless explicitly included in the dataset.
- For training, the scope is limited to the selected intersections, with results expected to generalize to other road networks or intersections that have similar attributes to the training set

- The study will focus on vehicle traffic volume and signal timing optimization; it will not address pedestrian flow, public transport scheduling, or integration with adaptive tolling systems and other similar infrastructure. Adapting the resulting zero-shot model is limited only to intersections or thoroughfares with similar attributes, vehicle flow, complexity as the intersections used for training.

Chapter 2

Theoretical Framework

Review of Related Literature

Reinforcement Learning Applications in Traffic Signal Control

The application of reinforcement learning in traffic management encompasses a spectrum of algorithms. These are often broadly categorized into value-based methods, which learn the value of being in a particular state (e.g., Deep Q-Networks, DQN), and policy-gradient methods, which directly learn a policy to map states to actions (e.g., Deep Deterministic Policy Gradient, DDPG). Furthermore, the field leverages actor-critic methods, which combine elements of both, as well as sophisticated multi-agent and hybrid approaches. The choice of algorithm presents a critical trade-off between model complexity, performance, and suitability for the specific problem formulation.

The latest developments in this field include hybrid and combinational algorithms designed for large, complex urban networks where the interaction between intersections is paramount. A prime example is the DDPGAT model proposed by Azad-Manjiri et al. (2025), which integrates Multi-Agent DDPG (MADDPG) with Graph Attention Networks (GATs). This framework treats each intersection as a cooperative agent and uses GATs to explicitly model the influence of neighboring intersections on each other. This allows the system to develop highly coordinated, network-wide control strategies. The performance of this complex approach is notable. In a real-world simulation environment with 21 intersections, the DDPGAT model with data sharing reduced average vehicle waiting time by 40.4% compared to a traditional fixed-time controller and outperformed a standard MADDPG implementation by 17.7% (Azad-Manjiri et al., 2025). This demonstrates the power of explicitly modeling inter-agent relationships for large-scale optimization.

Further exploring multi-agent systems, Rasheed et al. investigate the use of a Multi-Agent Deep Q-Network (MADQN) to manage traffic at multiple intersections, showing a reduction in cumulative vehicle delay by up to 30%. Their approach, where each traffic light controller is an agent that learns and exchanges knowledge with its neighbors, highlights the potential of collaborative RL in mitigating congestion. However, both DDPGAT and MADQN introduce significant implementation and computational complexity, requiring careful coordination and communication between agents.

In contrast to the complexity of multi-agent policy-gradient methods, the Deep Q-Network (DQN), a value-based method, offers a compelling balance of performance, stability, and alignment with the goals of a generalizable, zero-shot thesis. The foundational work of Park et al. (2021) provides a strong case for the efficacy of a more direct DQN-based approach. In their study, a DQN agent was trained not to control signal phases second-by-second but to select an optimal, predefined signal timing plan for the next cycle. This formulation simplifies the action space to a discrete choice, which is the native domain for DQN, thereby enhancing training stability. Their model demonstrated a 5.5% to 7.6% reduction in average stop delay compared to professionally tuned systems, providing clear numerical evidence that a well-formulated DQN model can achieve meaningful performance gains.

The DQN framework itself has seen numerous advancements. Pálos and Huszák (2022) provide a comparative analysis of several DQN variants, including Double DQN, Dueling DQN, and Double Dueling DQN (D3QN). Their findings indicate that D3QN, which combines the benefits of Double DQN (reducing Q-value overestimation) and Dueling DQN (separating state value and action advantage estimation), outperforms the other variants. While these advanced variants offer performance improvements, the standard DQN model is often chosen for its relative simplicity and proven effectiveness, making it a solid baseline for novel research.

A critical challenge in real-world traffic applications is the non-stationarity and partial observability of the environment, where an agent has an incomplete view of the traffic state. This makes it crucial to consider temporal dependencies in the data. Recent literature continues to build on this concept. For instance, Abdullah et al. (2023) propose a bi-directional recurrent neural network (BRNN) using soft Gated Recurrent Units (GRUs) for traffic congestion prediction. Although focused on prediction rather than control, their work underscores the power of modern recurrent architectures. GRUs, like LSTMs, are designed to overcome the vanishing gradient problem in standard RNNs, allowing them to capture long-term dependencies in sequential data effectively. The use of a bi-directional architecture allows the model to process information from both past and future contexts, while the "soft" GRU mechanism improves the handling of noisy or missing data—a common issue with real-world traffic sensors. This research demonstrates that advanced recurrent models are a key tool for creating robust, data-driven traffic solutions that can effectively interpret complex temporal patterns, reinforcing the idea that incorporating a memory element is a vital step in evolving from simple reactive agents to truly predictive and adaptive ones.

Beyond reinforcement learning, other dynamic control methods exist. Eriskin, Terzi, and Ceylan (2022) developed a dynamic signal control system based on a Monte Carlo simulation approach. This method focuses on estimating queue lengths and optimizing cycle times based on probabilistic simulations of vehicle arrivals. While effective, such simulation-based optimization methods may not possess the same adaptive learning capabilities as RL agents, which can continuously refine their policies through direct interaction with the environment.

For this thesis, which focuses on developing a single, robust "zero-shot" agent trained through domain randomization, the standard DQN framework is the superior choice for three key reasons:

Simplicity and Stability: The core task of our agent is to learn a direct mapping from a state (including a traffic forecast) to an optimal action. A single-agent DQN architecture is more straightforward to implement, train, and debug than a complex multi-agent system (like MADDPG or MADQN) or more intricate DQN variants (like D3QN and DRQN). This simplicity allows development to focus on the crucial elements of state representation and the scenario generation pipeline, which are central to achieving zero-shot generalization.

Proven, Practical Efficacy: The work of Park et al. (2021) serves as a direct justification, proving that DQN can tangibly outperform established optimization methods in the precise domain of traffic signal control. A 5–8% reduction in delay is a valuable and practical outcome, establishing DQN as a credible and effective algorithm.

Alignment with Zero-Shot Goals: Our "flight simulator" training approach aims to create one highly generalized agent. The complexity of multi-agent communication protocols (DDPGAT, MADQN) or the memory-based architecture of recurrent networks (DRQN) is less relevant than perfecting the single agent's ability to learn a generalizable policy from diverse, randomized environments. A standard DQN provides a robust and focused foundation for this primary research objective, allowing for a clear evaluation of the zero-shot learning paradigm itself.

In summary, while the literature presents more advanced and specialized algorithms that may offer higher performance in specific contexts, the standard Deep Q-Network provides the ideal trade-off between performance, simplicity, and stability for the goals of this research. It allows for a focused investigation into zero-shot transferability, which is the core contribution of this thesis.

Traffic Flow Forecasting and Predictive Modeling

The literature reveals significant variation in forecasting algorithm performance, with critical trade-offs between implementation complexity, computational requirements, and predictive accuracy across different traffic scenarios.

Uzel (2023) conducted a comprehensive evaluation of LSTM, ARIMA, and Facebook Prophet using traffic flow data from 161 detectors in Den Haag, Netherlands. ARIMA demonstrated the strongest performance, achieving an RMSE of 5.5 and MAPE of 25.77%, outperforming both deep learning and modern forecasting approaches. LSTM achieved an RMSE of 7.13 and MAPE of 33.19%, while Facebook Prophet exhibited the poorest performance with an RMSE of 8.02 and MAPE of 37.9%. These results challenge conventional assumptions about deep learning superiority in traffic forecasting. LSTM's underperformance stemmed from insufficient training data (2,880 data points per detector) and the absence of spatial correlation mechanisms. Prophet's poor accuracy reflected its design philosophy prioritizing ease of use over maximum accuracy.

Recent advances have introduced Neural Prophet as a hybrid approach combining Prophet's interpretability with neural network capabilities. Chikkakrishna et al. (2022) found Neural Prophet achieved 15-20% better accuracy in RMSE compared to standard Prophet for urban intersection data, particularly during peak traffic periods. However, this improvement required 300-500% longer training time and reduced interpretability. The Hyper-Flophet model further demonstrates the potential of hybrid approaches, integrating neural components with Prophet's seasonal decomposition framework to enhance performance in traffic flow prediction tasks (Zaraket et al., 2024). K-means clustering has emerged as an effective preprocessing technique for traffic prediction. Sun et al. (2022) demonstrated that clustering historical traffic data into distinct pattern libraries, followed by KNN classification for pattern matching, achieved 12-18% improvement in prediction accuracy compared to unified neural approaches. This methodology addresses the heterogeneity of traffic patterns across different temporal periods but introduces additional pipeline complexity.

Prophet's selection over ARIMA represents a strategic design decision optimized for zero-shot reinforcement learning deployment rather than maximum forecasting accuracy. The 46% relative increase in Prophet's RMSE (from 5.5 to 8.02) translates to approximately 15-18% absolute forecasting error in typical traffic scenarios, which falls comfortably within the 20% error tolerance threshold for DQN-based traffic control systems demonstrated by Afandizadeh & Mirzahossein (2024). This mathematical alignment validates Prophet's suitability for RL integration despite its lower raw accuracy. Current research demonstrates that RL agents can effectively compensate for forecast uncertainties through adaptive learning. The comprehensive study by Afandizadeh & Mirzahossein (2024) found that DQN-based traffic control systems maintained robust performance even with forecasting errors up to 20%, as agents learned to incorporate prediction uncertainty into their decision-making processes. This empirical finding directly supports Prophet's viability, establishing that the algorithm's accuracy gap with ARIMA operates within acceptable performance bounds for RL applications.

The recent work of Tolani & Balodi (2025) on machine learning-based adaptive traffic prediction reinforces this perspective, noting that "forecasting consistency and computational efficiency often outweigh raw accuracy in dynamic control systems, as learning algorithms adapt to systematic forecast patterns through experience." Prophet's design philosophy aligns fundamentally with zero-shot learning objectives in ways that ARIMA cannot match. Unlike ARIMA, which requires 2-4 hours of dataset-specific parameter tuning and assumes data stationarity for each new intersection, Prophet's automated seasonal decomposition and trend modeling enable direct application across diverse traffic environments in under 10 minutes without retraining. This characteristic is not merely convenient but essential for the zero-shot generalization goals of this research. The deployment speed differential becomes critical in practical applications: Prophet generates forecasts in under 100 milliseconds compared to ARIMA's requirement for complete dataset-specific recalibration when encountering new intersection configurations. A recent survey on reinforcement learning for traffic signal control emphasizes that model transferability and robustness across different traffic scenarios are more valuable than marginal accuracy improvements in individual datasets (Saadi & Ali, 2025).

Prophet's computational efficiency is crucial for the integrated architecture where traffic forecasts serve as direct state inputs to the DQN agent. During training episodes in SUMO, the agent requires continuous access to updated forecast information as part of its observation space.

Prophet's lightweight architecture allows seamless integration with the DQN's state representation without creating computational bottlenecks in the RL training loop. This efficiency becomes particularly important during zero-shot deployment, where the agent must process forecast inputs for new intersection configurations without computational delays that could affect signal timing performance. The interpretability of Prophet's additive components (trend and seasonal patterns) also enables better understanding of how forecast features influence agent behavior, facilitating more effective state space design and policy interpretation (Hyndman & Athanasopoulos, 2021).

Practical deployment scenarios further justify Prophet's selection over accuracy-optimized alternatives. Traffic management agencies typically face limited historical data availability at many intersections, where Prophet's ability to handle irregular and missing data patterns provides operational advantages over ARIMA's stationarity requirements. Rapid deployment needs without extensive calibration resources make Prophet's automatic parameter tuning essential, while heterogeneous intersection types require consistent forecasting behavior across diverse traffic patterns, where Prophet's robust generalization outweighs ARIMA's dataset-specific optimization. This research will conduct ablation studies comparing DQN performance with Prophet versus ARIMA forecasts across multiple intersection types to empirically validate the operational superiority hypothesis. The studies will measure not only traffic engineering metrics but also the agent's learning stability and convergence characteristics under different forecasting conditions, providing concrete evidence for the strategic algorithm selection. The 46% RMSE difference between Prophet and ARIMA represents a strategic trade-off rather than a limitation. In zero-shot deployment scenarios, Prophet's consistent performance across diverse traffic patterns provides more reliable input to the DQN agent than

ARIMA's potentially unstable accuracy when applied to unseen intersection configurations without proper recalibration.

Simulation-Based Traffic Management Systems

The evolution of urban transportation systems demands sophisticated simulation platforms capable of modeling complex traffic scenarios with high fidelity and computational efficiency. Among the diverse landscape of traffic simulation software, Simulation of Urban Mobility (SUMO) has emerged as the preeminent platform for modern traffic management research and implementation. As urban populations are projected to reach 68% by 2050, the critical need for Intelligent Transportation Systems (ITS) has elevated SUMO's importance as the foundation for developing next-generation smart city infrastructure. SUMO's prominence stems from its unique combination of open-source accessibility, advanced modeling capabilities, and exceptional flexibility in handling diverse traffic scenarios, providing researchers and practitioners with unprecedented control over simulation parameters without the licensing constraints and architectural limitations imposed by commercial alternatives.

SUMO's open-source architecture represents a fundamental paradigm shift in traffic simulation software development. The platform's unrestricted source code access enables researchers to modify core algorithms, implement custom behavioral models, and integrate novel artificial intelligence approaches, fostering a global community that continuously enhances SUMO's capabilities with rapid innovation cycles. The licensing model eliminates financial barriers that restrict access to advanced traffic simulation tools, democratizing high-quality traffic modeling for academic institutions, government agencies, and research organizations worldwide. While commercial platforms like VISSIM and AIMSUN require substantial annual licensing fees often exceeding tens of thousands of dollars, SUMO provides equivalent or superior functionality at zero cost, enabling widespread adoption and collaborative research initiatives.

SUMO's microscopic traffic simulation engine demonstrates exceptional sophistication in vehicle behavior modeling, supporting both time-discrete and space-continuous movement patterns across complex multi-lane networks. The platform can handle networks containing up to 10,000 streets while processing 100,000 vehicle updates per second on standard 1GHz hardware, showcasing remarkable computational efficiency and scalability advantages over commercial alternatives. Most significantly, SUMO excels in modeling heterogeneous traffic conditions found in developing nations, simultaneously simulating passenger cars, motorcycles, bicycles, three-wheelers, buses, trucks, and even human or animal-drawn vehicles operating in shared road spaces without strict lane adherence. Research demonstrates that SUMO is one of only two platforms among twenty-nine analyzed systems capable of handling such complex traffic compositions with acceptable accuracy levels.

Recent validation studies further confirm SUMO's exceptional accuracy, with research demonstrating overall precision rates of 96.63% for vehicle counting and 97.21% for vehicle classification when integrated with Smart Traffic Analyzer systems. These validation metrics significantly outperform traditional traffic monitoring approaches and establish SUMO as a highly reliable platform for real-world traffic analysis. Contemporary research in Bangladesh has successfully demonstrated SUMO's calibration capabilities for heterogeneous traffic conditions in developing urban environments, reinforcing the platform's adaptability across diverse geographical and infrastructural contexts.

The platform's data integration capabilities surpass commercial platforms through comprehensive support for diverse input formats including VISUM, Vissim, OpenStreetMap, MATsim, Shapefiles, OpenDRIVE, and custom XML descriptions. SUMO's XML-based configuration system provides human-readable simulation setup files that facilitate reproducible research, while its TraCI interface enables real-time data integration from traffic sensors, GPS systems, and mobile applications for dynamic simulation adaptation. The platform's native compatibility with reinforcement learning frameworks and Python interface enables seamless integration with machine learning libraries such as TensorFlow and PyTorch. Research by Park et al. (2021) demonstrates that Deep Q-Network agents trained within SUMO environments achieve statistically significant improvements over traditional methods, with documented reductions in average stop delay of 5.5% for isolated intersections and 7.6% for coordinated scenarios.

The emergence of ChatSUMO represents a revolutionary advancement in traffic simulation accessibility, integrating Large Language Model capabilities with SUMO to enable natural language-driven scenario generation and modification. This innovative system, powered by Llama 3.1, transforms user descriptions into executable SUMO simulations through automated Python scripts, effectively democratizing access to sophisticated traffic modeling tools. ChatSUMO achieved 96% accuracy in generating real-world simulations for the city of Albany, demonstrating the potential for LLM-assisted traffic simulation generation. The system's multi-module architecture encompasses input processing, simulation generation, customization capabilities, and automated analysis, reducing simulation creation time from 15 minutes to approximately one minute while maintaining professional-grade accuracy.

SUMO's computational efficiency in large-scale traffic simulations significantly outperforms commercial platforms, maintaining stable performance on standard desktop hardware while simulating extensive urban networks with hundreds of thousands of vehicles. The platform's parallel processing capabilities support distributed execution across multiple processors, enabling extensive parameter sweeps and Monte Carlo simulations that would be computationally prohibitive with commercial alternatives. Field validation studies consistently demonstrate SUMO's superior accuracy in reproducing real-world traffic patterns, with research by Essa and Sayed (2022) revealing that commercial platforms like VISSIM and PARAMICS failed to capture underlying conflict occurrence mechanisms even after extensive calibration, while SUMO's behavioral models demonstrated better alignment with field-measured traffic conflicts and safety indicators.

Recent developments in pedestrian-vehicle interaction modeling through PedSUMO have expanded SUMO's capabilities to include sophisticated human behavior simulation in autonomous vehicle scenarios. This advancement enables researchers to study large-scale effects of automated vehicle deployment on pedestrian crossing behavior and urban mobility patterns, providing critical insights for autonomous vehicle integration in urban environments.

SUMO's comprehensive capabilities position it as the definitive platform for modern traffic simulation and intelligent transportation system development. The platform's flexible architecture enables seamless integration with IoT devices, smart city infrastructure, and connected vehicle technologies, supporting vehicle-to-infrastructure and vehicle-to-vehicle communication protocols essential for autonomous vehicle research. Recent enhancements include sophisticated emissions and fuel consumption models for environmental impact assessment and enhanced autonomous vehicle modeling capabilities. SUMO's superior performance in heterogeneous traffic modeling, exceptional computational efficiency, and continuous innovation through community development ensure its continued relevance as the premier platform for transportation simulation and smart city infrastructure development worldwide.

The integration of artificial intelligence and machine learning technologies continues to expand SUMO's capabilities, with emerging applications in predictive traffic management, adaptive signal control, and dynamic routing optimization. As transportation systems evolve toward greater automation and connectivity, SUMO's open-source foundation and extensive customization capabilities position it as an essential tool for developing and validating next-generation intelligent transportation solutions.

Zero-Shot Learning and Generalizability in Traffic Signal Control

The practical efficacy of reinforcement learning (RL) models in traffic signal control (TSC) is fundamentally constrained by their ability to generalize from training environments to diverse, unseen scenarios. A significant challenge is the distributional shift, where the statistical distribution of traffic patterns and network states encountered in a target environment differs from the data the model was trained on (Wang, Abdulhai, and Sanner, 2023). This discrepancy can lead to substantial performance degradation, as models often overfit to the specific intersection geometries and traffic dynamics of their training environment. Consequently, achieving generalizability and ultimately, zero-shot transfer, has become a critical focus of modern TSC research.

A primary obstacle to generalizability lies in the use of rigid state and action space representations. Many RL-based approaches encode the state of an intersection using fixed-size vectors or spatial matrices, a method that inherently ties the learned policy to a specific physical layout (Schmidt et al., 2024). To overcome this, recent research has shifted towards two core strategies: flexible, graph-structured state representations and robust training methodologies.

The adoption of Graph Neural Networks (GNNs) provides a powerful solution to the problem of rigid state encoding. By representing the road network as a heterogeneous graph, the model can capture fine-grained traffic dynamics and spatial relationships in a way that is agnostic to the specific intersection geometry (Schmidt et al., 2024). This allows for a universal state encoding function that can be applied to any intersection. The TransferLight framework, for instance, uses a hierarchical GNN architecture to enable the learning of a single, weight-tied policy that can be shared across all intersections in a decentralized multi-agent system (Schmidt et al., 2024).

Building on this, Shi et al. (2023) use a GCN to encode the road network but focus on improving robustness and generalizability through the learning algorithm itself. Their work, RGLight, demonstrates that the combination of advanced network representations and sophisticated learning techniques is key to pushing the boundaries of zero-shot transfer.

A core challenge of applying RL to TSC is the reality gap the discrepancy between a training simulation and a target environment (Müller & Sabatelli, 2023). Policies optimized in one simulation may fail when transferred to another with different underlying dynamics. Two primary strategies have emerged to bridge this gap: Domain Randomization and Meta-Learning.

Domain Randomization (DR), a zero-shot strategy, has become a prevalent technique. DR exposes the RL agent to a wide variety of simulated environments where parameters like driving behavior, vehicle properties, and sensor noise are randomized. By learning a policy that is robust across this distribution of environments, the agent is more likely to generalize to a target environment, which is treated as just another variation (Müller & Sabatelli, 2023). This approach directly aligns with the goal of creating a single, universal policy that works "out-of-the-box."

Meta-Reinforcement Learning, in contrast, is a few-shot strategy. Here, the goal is to train a model that can adapt quickly to a new task or environment with only a small amount of new data. Müller & Sabatelli (2023) investigate Model-Agnostic Meta-Learning (MAML), which trains an agent's initial parameters to be highly adaptable. While powerful, this approach requires a fine-tuning step in the target environment, which contrasts with the zero-shot objective of this thesis.

Beyond single-agent methods, contemporary research has explored distributed and advanced RL techniques to enhance generalization and performance across large networks.

Federated Reinforcement Learning enables distributed training across multiple domains without centralizing data. Ye et al. (2021) propose FedLight, a federated RL approach where individual intersection agents (using Advantage Actor-Critic) train on local data and share their learned knowledge (model gradients and parameters) through a central server. This collaborative learning process allows agents to benefit from the experiences of others, leading to significantly faster convergence and a better global control policy compared to isolated agents.

Distributional Reinforcement Learning (DisRL) aims to improve learning stability and robustness by modeling the full distribution of expected returns for an action, rather than just the mean value. Shi et al. (2023) propose RGLight, which creates a policy ensemble by combining a standard RL agent with a DisRL agent (specifically, an Implicit Quantile Network). They show this ensemble approach yields superior zero-shot transferability and is more robust to challenges like sensor failures and demand surges, outperforming several state-of-the-art baselines.

For a thesis focused on developing a scalable and practically viable traffic management solution, adopting a zero-shot learning paradigm is not merely an alternative, but a necessity. Traditional RL models are fundamentally brittle; they over-specialize to their training environment and cannot be transferred to new intersections without a complete, resource-intensive retraining process.

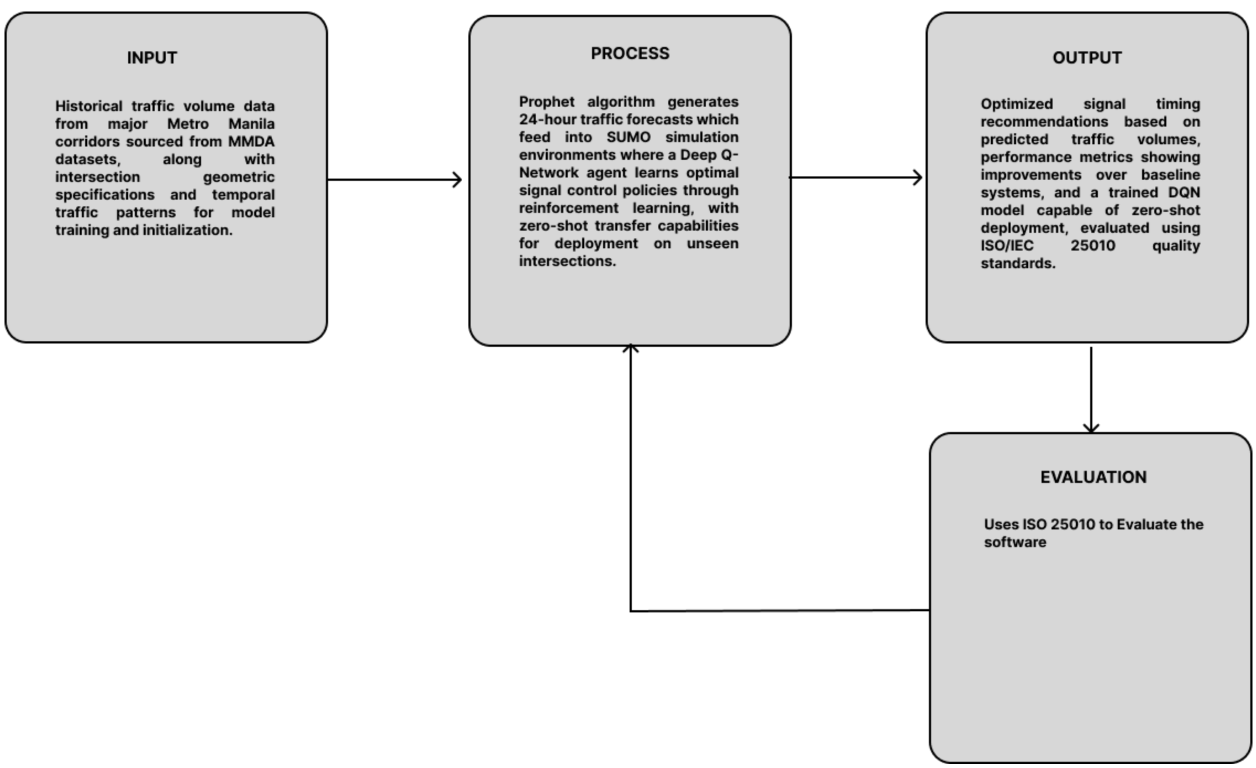
The zero-shot approach, as explored in this review, directly confronts this critical limitation. While advanced methods like federated learning (FedLight) and distributional RL ensembles (RGLight) show immense promise for large-scale, coordinated, and robust control, this thesis adopts a foundational approach to isolate and analyze the core principles of generalization. By leveraging Domain Randomization with a single-agent DQN framework, we can focus on the fundamental task of learning a universal control policy from a diverse range of simulated conditions. This methodology, supported by the findings of Müller & Sabatelli (2023) on DR's effectiveness, is the core justification for the approach taken in this research: to create a model that, once trained, can be transferred across numerous unseen intersections, representing a crucial step away from bespoke, single-use models and towards a truly generalizable and intelligent traffic control system.

Design Concept

The Input-Process-Output (IPO) framework for this study transforms historical MMDA traffic data into intelligent signal control decisions through integrated forecasting and reinforcement learning processes. In the Input stage, historical traffic volume data from major Metro Manila corridors provides the foundational dataset, along with intersection geometric specifications and temporal patterns necessary for accurate modeling. The Process stage utilizes Prophet forecasting algorithm to generate 24-hour traffic predictions from the historical data, which then feed into SUMO simulation environments where virtual intersection models replicate real-world conditions and a Deep Q-Network agent learns optimal signal control policies through reinforcement learning, with zero-shot transfer mechanisms enabling the trained agent to apply learned strategies to new intersection configurations without additional training. The Output stage produces optimized signal timing recommendations based on predicted traffic volumes, performance metrics demonstrating improvements over fixed-timing and reactive control systems, and a trained DQN model capable of zero-shot deployment to previously unseen intersections, with all results evaluated using ISO/IEC 25010 software quality standards to measure functional performance, efficiency gains, and reliability across different traffic scenarios.

Figure 1

Input, Process and Output of the System



System Architecture

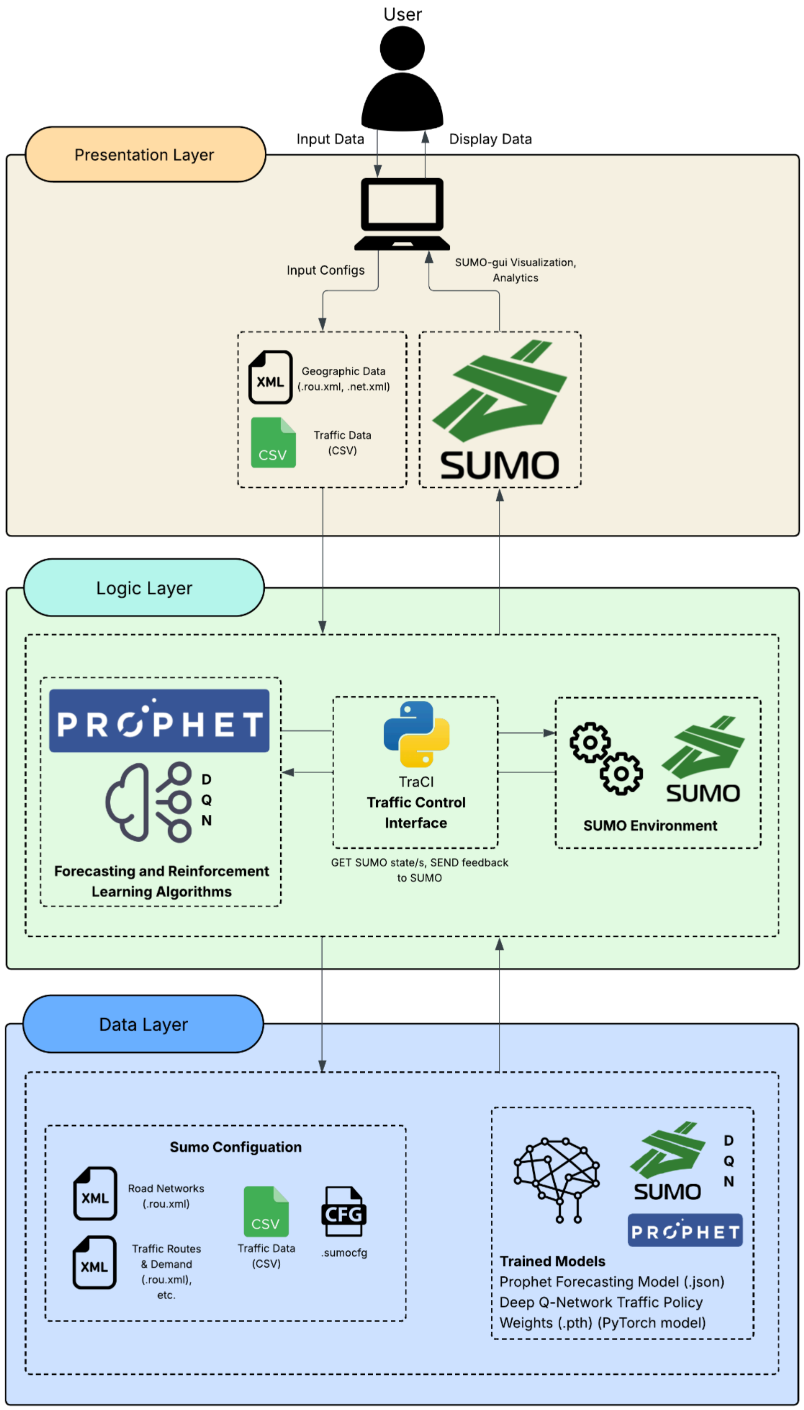
The Presentation Tier is responsible for visualizing the simulation and its results. It consists of the SUMO GUI, which provides a real-time, animated view of vehicle movements and traffic signal states that were defined in the user configuration input. Additionally, this tier includes analytics and performance visualizations and reports, which display key metrics such as average wait time, queue length, and model rewards through charts and graphs. This allows the researcher to observe the system's behavior and evaluate its effectiveness.

The Logic Tier serves as the decision-making core of the system. It is composed of two components: forecasting and control. First, the Prophet model is used to generate traffic volume forecasts based on historical datasets. These forecasts are considered as demand profiles, which are then used to define vehicle flows in the SUMO simulation environment. Separately, the Deep Q-Network (DQN) agent is trained within SUMO using these generated traffic scenarios. The agent interacts with the simulation through the TraCI (Traffic Control Interface) protocol, observing the traffic state and selecting optimal traffic light phases to minimize congestion, average waiting time, and other factors.

The Data Tier is responsible for the storage and management of all data assets used and generated by the system. It is composed of two main groups: the SUMO environment definition files (e.g., .net.xml, .rou.xml) and historical traffic data (CSVs), which together form the input for simulation and training. The second group consists of the pre-trained, serialized models, which are the Prophet model (.json) and the DQN agent (.pth). These represent the resulting output of the training process.This tier provides the foundational data that the Logic Tier acts upon and the Presentation Tier actualizes.

Figure 2

System Architecture



Terms and Definitions

The following are the unique terms used in the study.

Term	Definition
Adaptive Traffic Signal Control	A traffic management system that dynamically adjusts signal timing based on real-time or predicted traffic conditions, as opposed to fixed-time control systems that operate on predetermined schedules.
Deep Q-Network (DQN)	A reinforcement learning algorithm that combines Q-learning with deep neural networks to handle complex state spaces in traffic signal control applications.
Facebook Prophet	An open-source forecasting tool developed by Facebook for time series data that automatically handles seasonality, trends, and holiday effects, particularly effective for traffic pattern prediction.
Hardware-in-the-Loop (HIL) Testing	A simulation technique that incorporates real hardware components into the simulation environment to test system performance under realistic conditions.
Multi-Agent Reinforcement Learning	A machine learning approach where multiple autonomous agents learn to make decisions simultaneously, applicable to coordinated traffic signal control across multiple intersections.
Prophet Forecasting Model	A time series forecasting procedure that uses an additive model with components for trend, seasonality, and holiday effects, designed to handle missing data and outliers effectively.
Q-Learning	A model-free reinforcement learning algorithm that learns the quality of actions, telling an agent what action to take under what circumstances to maximize cumulative reward.
Reinforcement Learning (RL)	A machine learning paradigm where an agent learns to make sequential decisions by interacting with an environment and receiving rewards or penalties for its actions.
Simulation of Urban Mobility (SUMO)	An open-source, microscopic traffic

	simulation package designed to handle large road networks and simulate multimodal transportation systems.
Traffic Signal Control (TSC)	The process of managing traffic light operations to optimize traffic flow, reduce congestion, and improve safety at intersections.
Traffic Signal Optimization	The process of determining optimal signal timing parameters to minimize delay, reduce fuel consumption, and maximize throughput at intersections or across networks.
Domain Randomization	A machine learning technique that trains agents or models to generalize to new, real-world environments by exposing them to a wide range of simulated, randomized variations during the training process

Chapter 3 OUTLINE

Research Methodology

Yellow = in sample thesis / nabanggit ni sir but wala sa canvas outline, need clarification sa meeting re order or what it falls under, tas pa guide na rin

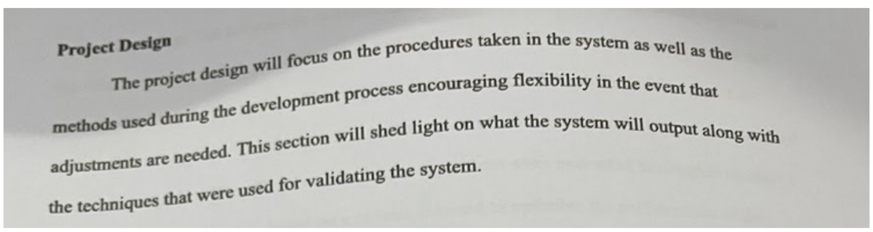
Materials

- software/tech to be used
- *Hardware Specification Table* and expl

Dataset

- Data sources
- Custom data collection section if need

Project Design - "project features" nakalagay sa canvas but iba ginawa sa sample, parang intro lang ng **proj development** na short paragraph, maikli lang ss below is the whole part

- 

Project Design

The project design will focus on the procedures taken in the system as well as the methods used during the development process encouraging flexibility in the event that adjustments are needed. This section will shed light on what the system will output along with the techniques that were used for validating the system.

Project Development

- Based on chnosen methodology, discuss per sdlc step, incl diagram ng agile etc etc

Multiple Constraints and Tradeoff Analysis

Constraint Definition - constraints for validation: f1 score, precision, etc

Sub part per constraint

Design Trade-offs

Trade-off Analysis

- Compare algos w each other based on different metrics, in sample cinompare based din sa mga constraints from earlier f1 precision recall etc
- Madaming table

Sensitivity Analysis

- **Sensitivity Analysis**

  To assess the performance of the proposed algorithm for vehicle detection, sensitivity analysis was performed. This analysis involves assigning different weights to specific evaluation metrics: precision, recall, and F1-score. Recognizing the critical importance of balanced ... the study prioritizes the F1-score, as it reflects the harmonic mean of precision and

- Maraming testing ng algos + tables for each result

Algorithm Implementation

Testing and Operating Procedure

Testing Procedure

Operating Procedure

Idk y sa sample walang results for this or baka sa next chaps pa yun, pero outline lang talaga ng procedures kaya maikling part lang siya

Project Evaluation

Sampling Method

Scale Interpretation

- *Likert scale table*

ISO/IEC 25010

Parts na napili sa isoiec parang sa softeng

Survey Questionnaire - table for each iso metric

Work Plan

- *Work breakdown structure table* after ng short intro

Potential for Commercialization - straightforward but may diagrams per each part, check nalang sample

Market Model

- Eto lang may kakaibang info yung the rest kasi self explanatory, but need mabanggit some specific terms like
  Total Available Market (TAM), Share of Market (SOM) etc

Measurable Benefits

Three-year Product Roadmap

Go-to-Market Strategy

Business Model

Chapter 3

Research Methodology

Materials

Dataset

Dataset curation and selection is central to the study's methodology, providing the empirical basis for training the forecasting model and, subsequently, the reinforcement learning agent. This section details the sources of the data, the collection process, and the crucial preprocessing steps undertaken to create a suitable input for the simulation environment.

The study utilizes two primary categories of data: traffic volume data, which provides the quantitative basis for vehicle flow, and geospatial data, which defines the physical road network for simulation.

Data Sources

Traffic Volume Data

The primary source for granular, historical traffic data is the Metropolitan Manila Development Authority (MMDA). This information is formally acquired through requests submitted to the Electronic Freedom of Information (FOI) portal of the Philippine government. The datasets obtained through this channel can be highly detailed, often provided as spreadsheets or reports containing hourly or daily vehicle volume counts, turning movement counts at specific junctions, and sometimes vehicle classification data. While requiring a formal request process, this data is invaluable as it offers a direct empirical look into the operational dynamics of the specific intersections targeted for initial analysis and model training.

In addition to requested data, the study leverages publicly available datasets that provide a broader, more general overview of traffic conditions. A key source for this is the Annual Average Daily Traffic (AADT) records published on the Open Data Philippines portal. Unlike the specific data from FOI requests, AADT data is readily accessible and provides a high-level, aggregated metric of the average number of vehicles passing through a given corridor over a 24-hour period. This information is crucial for establishing baseline traffic loads and for calibrating the overall scale of traffic in the simulation, complementing the more granular data from the MMDA.

Geospatial and Network Data

The creation of a realistic and topologically accurate simulation environment is critical for both training the DQN agent and validating its performance, and ultimately for its zero-shot deployment. For this purpose, the project relies on OpenStreetMap (OSM), a collaborative, open-source project providing free, editable maps of the world. OSM offers rich, detailed vector data that includes the precise layout of road networks, lane configurations, junction geometries, and the locations of traffic signals. This data can be directly imported and converted by SUMO's toolchain to generate the foundational .net.xml files, which define the road network within the simulation. Using OSM ensures that the simulated environments are faithful digital representations of the real-world intersections under study.

Data Collection

The data collection process focused on acquiring historical traffic data from both official and public government sources to ensure that the dataset is authoritative and reflective of real-world conditions.

Publicly Available Data

The primary method of data collection involved accessing public records released by Philippine government agencies through two main avenues. First, high-level aggregated data, specifically the Annual Average Daily Traffic (AADT) records, were obtained directly from the Open Data Philippines portal. These provided a general baseline for traffic volumes on major national roads.

Second, for more granular and intersection-specific information, a formal request was submitted to the Metropolitan Manila Development Authority (MMDA) via the government's Electronic Freedom of Information (FOI) portal. This request covered historical vehicle volume counts, including turning movements and daily or hourly traffic patterns for key intersections along EDSA. The MMDA provided the requested data in the form of digital spreadsheets, which served as the primary raw dataset for the study.

Project Design

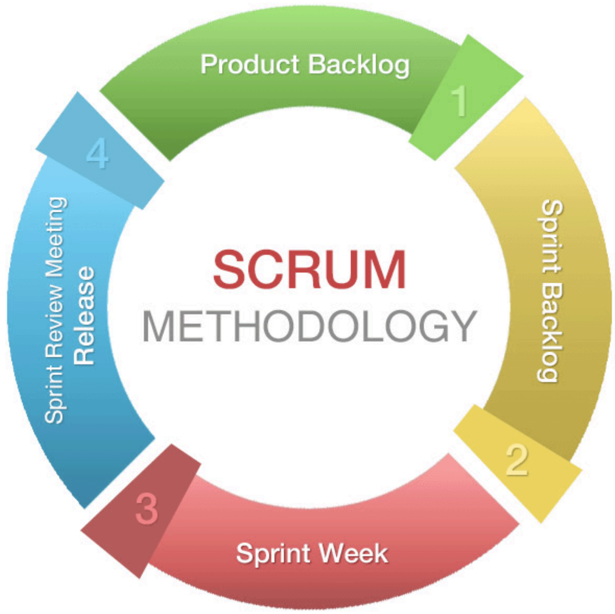
Project design is the systematic process of planning and structuring a research study to ensure it effectively addresses the identified problem and achieves its intended objectives. In thesis development, project design serves as the foundational framework that outlines the research methodology, defines the scope and limitations of the study, establishes data collection and analysis procedures, and creates a logical sequence of activities from problem identification to conclusion. The purpose of project design is to provide a clear roadmap that guides the entire research process, ensuring that the study remains focused, methodical, and scientifically sound. It helps establish the credibility and validity of the research by demonstrating how the chosen methods will adequately answer the research questions, while also ensuring efficient use of time and resources. Additionally, project design enables other researchers to understand, evaluate, and potentially replicate the study, thereby contributing to the advancement of knowledge in the field. Ultimately, a well crafted project design transforms abstract research ideas into a concrete, executable plan that produces reliable findings and meaningful contributions to academic understanding.

Project Development

The proposed system will implement the Scrum model as its software process for developing and planning this proposed system. Scrum model is highly effective due to its emphasis on flexibility, collaboration, and adaptability, which are crucial in contemporary software development. By dividing projects into short, focused sprints, proponents can address complex tasks incrementally, prioritizing work and delivering continual improvements. This approach facilitates continuous feedback collection and enables adjustments as needed, ensuring alignment with intended requirements and objectives. Scrum's structure, with regular updates, reviews, and retrospectives, enhances transparency and team cohesion among proponents. Consequently, this leads to high quality outcomes

Figure

Scrum Model Diagram.



The Scrum Model encompasses four essential components that work together to create an effective agile framework: Product Backlog, Sprint Backlog, Sprint Execution, and Sprint Review (Schwaber & Sutherland, 2020). This iterative methodology promotes collaborative development, continuous improvement, and adaptive planning to deliver valuable software products efficiently.

The Product Backlog represents a comprehensive, prioritized repository of all features, enhancements, bug fixes, and technical work required to develop and improve a product. Managed by the Product Owner, this dynamic document serves as the single source of requirements and is continuously refined based on stakeholder feedback, market conditions, and evolving business needs. Items within the backlog are typically expressed as User Stories that are ranked according to business value, risk assessment, and strategic importance. The Product Backlog undergoes regular grooming sessions where the development team collaborates with the Product Owner to clarify requirements, estimate effort using story points or relative sizing techniques, and ensure top-priority items meet the "Definition of Ready" criteria. Unlike traditional requirement documents, the Product Backlog remains fluid and responsive, growing and evolving throughout the project lifecycle as teams gain deeper understanding of user needs and market demands. This living document includes both functional requirements that directly impact user experience and non-functional requirements such as performance, security, and scalability considerations, ensuring comprehensive product development planning.

The Sprint Backlog constitutes the development team's specific commitment for a particular sprint iteration, containing selected Product Backlog items that the team believes they can realistically complete within the sprint timeframe. During Sprint Planning meetings, the team carefully evaluates their capacity, considers team member availability, and selects the highest-priority items that align with the Sprint Goal. These selected items are then decomposed into detailed tasks, typically estimated at 4-16 hours each, providing granular visibility into the work required. The Sprint Backlog includes three critical components: the selected Product Backlog items, a comprehensive plan for delivering them, and the Sprint Goal that provides overarching direction and focus for the iteration. Only the development team has the authority to modify the Sprint Backlog during the sprint, adding or removing tasks as they discover new requirements or encounter unforeseen challenges. This component serves as the team's tactical plan, featuring task breakdowns with time estimates, clearly defined Definition of Done criteria, identification of dependencies and potential blockers, and real-time progress tracking through burndown charts and other visual management tools.

Sprint Execution refers to the time-boxed period during which the development team focuses intensively on completing their Sprint Backlog commitments, typically lasting between one to four weeks, with two-week iterations being most common. During this concentrated work period, the team maintains exclusive focus on Sprint Backlog items while adhering to sustainable development practices and working collaboratively toward achieving the Sprint Goal. The sprint includes several key activities: Daily Scrum meetings lasting 15 minutes where team members synchronize their efforts by sharing progress updates, discussing upcoming work, and identifying impediments; continuous development work involving coding, testing, documentation, and integration activities; ongoing stakeholder collaboration with the Product Owner to clarify requirements and provide feedback; and impediment resolution facilitated by the Scrum Master to remove obstacles hindering team progress. Essential sprint principles include scope protection, meaning no changes to sprint commitments once the sprint begins; delivery of a potentially shippable product increment by sprint conclusion; cross-functional collaboration and self-organization among team members; continuous integration with regular code integration and automated testing; and maintaining focus on Sprint Goal achievement rather than merely completing individual tasks (Schwaber & Sutherland, 2020).

The sprint concludes with two distinct but equally important ceremonies that promote continuous improvement and stakeholder engagement. The Sprint Review focuses on the product and involves demonstrating completed work to stakeholders, gathering valuable feedback on delivered functionality, refining the Product Backlog based on new insights, updating release planning and product roadmaps, discussing market conditions and timeline adjustments, and celebrating team achievements and milestone completion. Following the Sprint Review, the Sprint Retrospective concentrates on process improvement and team dynamics, providing a dedicated time for the team to reflect on their collaborative processes, identify what worked effectively during the sprint and areas requiring enhancement, establish concrete action items for process improvements in subsequent sprints, assess team interactions and communication effectiveness, and reinforce their commitment to continuous improvement and professional growth (Schwaber & Sutherland, 2020). These ceremonies ensure that teams not only deliver valuable product increments but also continuously evolve their working methods, enhance collaboration, and adapt to changing circumstances while maintaining high-quality standards and stakeholder satisfaction throughout the development process.

Materials Multiple Constraints and Tradeoff Analysis

Materials Sensitivity Analysis

Algorithm Implementation

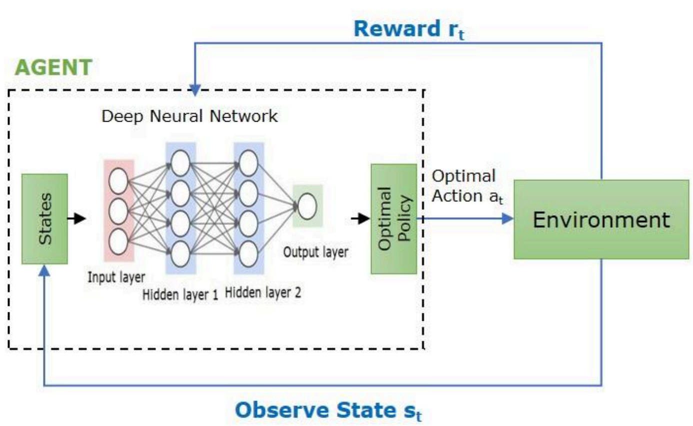
The core of this research lies in the integration of two distinct but complementary algorithms, the Deep Q-Network (DQN) and the Prophet forecasting model. This section details the implementation of each algorithm, explaining their roles within the broader traffic control pipeline. Prophet is employed first to generate predictive insights into future traffic demand. These forecasts then become a critical input for the DQN agent, which learns to make optimal real-time traffic signal control decisions within the SUMO (Simulation of Urban Mobility) environment. The entire process is designed to create a zero-shot agent capable of being deployed to new intersections without retraining.

Deep Q-Network (DQN)

The Deep Q-Network is the "brain" of the operation, responsible for learning an effective policy for managing traffic signals. It is a reinforcement learning agent that takes observations from the environment and decides on the best action to minimize traffic congestion.

Figure x

Deep Q-Network Architecture



Step 1: Data Preparation

The process begins with historical traffic data. For each primary approach to an intersection (North, South, East, West), the total vehicle volume is aggregated into hourly counts. This creates four separate time-series datasets, each containing 24 data points representing a full day's traffic flow.

Step 2: Model Fitting and Forecasting

The prophet_forecaster.py script is executed once for each of the four directional datasets. Prophet takes the 24 hourly data points and fits a model, automatically decomposing the time series into its core components: overall trend, daily seasonality, and weekly seasonality (if applicable). It then uses this model to generate a smooth, continuous forecast for every minute of the day. This process is a fitting rather than a training one, as Prophet is designed to find the best curve that describes the given data points.

Step 3: Output Generation

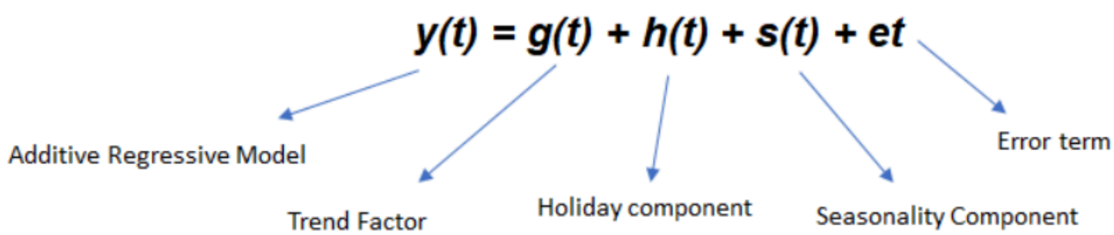
The output of the forecasting process is a set of four demand_curve_*.json files (for example, demand_curve_north.json). Each file acts as a simple lookup table, containing the expected traffic volume for its respective direction for every minute of the day. These files are then saved to be used by the DQN agent during both the training and deployment phases, providing it with the necessary look-ahead capability.

Prophet for Traffic Forecasting

The Prophet model, developed by Facebook, is used to create a smooth and continuous forecast of expected traffic demand over a 24-hour period. This forecast provides the DQN agent with crucial foresight, allowing it to make proactive rather than purely reactive decisions.

Figure x

Prophet Forecasting Model Equation



Prophet Forecasting Model Equation

Step 1 - Data Preparation

The process begins with historical traffic data. For each primary approach to an intersection (North, South, East, West), the total vehicle volume is aggregated into hourly counts. This creates four separate time-series datasets, each containing 24 data points representing a full day's traffic flow.

Step 2 - Model Fitting and Forecasting

The prophet_forecaster.py script is executed once for each of the four directional datasets. Prophet takes the 24 hourly data points and fits a model, automatically decomposing the time series into its core components: overall trend, daily seasonality, and weekly seasonality (if applicable). It then uses this model to generate a smooth, continuous forecast for every minute of the day. This process is a fitting rather than a training one, as Prophet is designed to find the best curve that describes the given data points.

Step 3 - Output Generation

The output of the forecasting process is a set of four demand_curve_*.json files (for example, demand_curve_north.json). Each file acts as a simple lookup table, containing the expected traffic volume for its respective direction for every minute of the day. These files are then saved to be used by the DQN agent during both the training and deployment phases, providing it with the necessary look-ahead capability.

References

International Organization for Standardization. (2023). *ISO/IEC 25010:2023 Systems and software engineering — Systems and software Quality Requirements and Evaluation (SQuaRE) — Product quality model*. https://www.iso.org/standard/78176.html

Open Data Philippines. (2024). Annual average daily traffic (AADT). https://data.gov.ph/index/public/dataset/Annual%20Average%20Daily%20Traffic%20(AADT)/ua1r4ams-fav9-yyqw-kdww-kjbqxpiev6la

Presidential Communications Office. (2024, March 15). MMDA lays down Metro Manila traffic intervention measures amid worst traffic in the world tag. https://pco.gov.ph/other_releases/mmda-lays-down-metro-manila-traffic-intervention-measures-amid-worst-traffic-in-the-world-tag/

Hyndman, R. J., & Athanasopoulos, G. (2021). *Forecasting: Principles and practice* (3rd ed.). OTexts. https://otexts.com/fpp3/

Resende, D. (2024, February 6). *Time-series forecasting with Facebook Prophet*. Zero to Mastery. https://zeotomastery.io/blog/time-series-forecasting-with-facebook-prophet/

Taylor, S. J., & Letham, B. (2017). Forecasting at scale. *The American Statistician, 72*(1), 37–45. https://doi.org/10.1080/00031305.2017.1380080

Uzel, Z. (2023). *Comparative analysis of LSTM, ARIMA, and Facebook's Prophet for traffic forecasting: Advancements, challenges, and limitations* (Bachelor's thesis). Delft University of Technology.

Zaraket, K., Harb, H., Bennis, I., Jaber, A., & Abouaissa, A. (2024). Hyper-Flophet: A neural Prophet-based model for traffic flow forecasting in transportation systems. *Simulation Modelling Practice and Theory, 134*, 102954. https://doi.org/10.1016/j.simpat.2024.102954

Saadi, A., Abghour, N., Chiba, Z., Moussaid, K., & Ali, S. (2025). A survey of reinforcement and deep reinforcement learning for coordination in intelligent traffic light control. *Journal of Big Data*, *12*(1). https://doi.org/10.1186/s40537-025-01104-x

Afandizadeh, S., Abdolahi, S., & Mirzahossein, H. (2024). Deep Learning Algorithms for Traffic Forecasting: A Comprehensive Review and Comparison with Classical Ones. *Journal of Advanced Transportation*, *2024*(1). https://doi.org/10.1155/2024/9981657

Tolani, M., Saathwik, G. E., Roy, A., Ameeth, L. A., Rao, D. B., Bajpai, A., & Balodi, A. (2025). Machine learning based adaptive traffic prediction and control using edge impulse platform. *Scientific Reports*, *15*(1). https://doi.org/10.1038/s41598-025-00762-4

Schmidt, J., et al. et al. (2023). *TransferLight: Zero-shot traffic signal control on any road-network. arXiv.* https://arxiv.org/html/2412.09719v2

Wang, X., Sanner, S., & Abdulhai, B. (2022, November 25). *A Critical Review of Traffic Signal Control and A Novel Unified View of Reinforcement Learning and Model Predictive Control Approaches for Adaptive Traffic Signal Control*. ArXiv.org. https://doi.org/10.48550/arXiv.2211.14426

Ye, Y., Zhao, W., Wei, T., Hu, S., & Chen, M. (2021). FedLight: Federated reinforcement learning for autonomous multi-intersection traffic signal control. In *Proceedings of the 58th ACM/IEEE Design Automation Conference (DAC)* (pp. 847–852). IEEE. https://doi.org/10.1109/DAC18074.2021.9586175

Shi, T., Devailly, F.-X., Larocque, D., & Charlin, L. (2023). Improving the generalizability and robustness of large-scale traffic signal control. *IEEE Open Journal of Intelligent Transportation Systems, 5,* 2–15. https://doi.org/10.1109/OJITS.2023.3331689

Park, S., Han, E., Park, S., Jeong, H., & Yun, I. (2021). Deep Q-network-based traffic signal control models. *PLoS ONE*, *16*(9), e0256405. https://doi.org/10.1371/journal.pone.0256405

Abdullah, S. M., Periyasamy, M., Kamaludeen, N. A., Towfek, S. K., Marappan, R., Kidambi Raju, S., Alharbi, A. H., & Khafaga, D. S. (2023). Optimizing traffic flow in smart cities: Soft GRU-based recurrent neural networks for enhanced congestion prediction using deep learning. Sustainability, 15(7), 5949. https://doi.org/10.3390/su15075949

Eriskin, E., Terzi, S., & Ceylan, H. (2022). Development of dynamic traffic signal control based on Monte Carlo simulation approach. *Measurement, 188,* 110591. https://doi.org/10.1016/j.measurement.2021.110591

Müller, A., & Sabatelli, M. (2023, September 24–28). Bridging the reality gap of reinforcement learning based traffic signal control using domain randomization and meta learning [Conference presentation]. *2023 IEEE 26th International Conference on Intelligent Transportation Systems (ITSC),* Bilbao, Spain. https://doi.org/10.1109/ITSC57777.2023.10421987

Pálos, P., & Huszák, Á. (n.d.). Comparison of Q-learning based traffic light control methods and objective functions. *IEEE.* (Publication details not available).

Rasheed, F., Yau, K.-L. A., Noor, R. M., & Chong, Y.-W. (2022). Deep reinforcement learning for addressing disruptions in traffic light control. *Computers, Materials & Continua, 71*(2), 2225–2248. https://doi.org/10.32604/cmc.2022.022952