TRIBHUVAN UNIVERSITY

INSTITUTE OF ENGINEERING

PULCHOWK CAMPUS

THESIS NO.: T02/078

Calibration of VISSIM Parameters for Modeling Heterogeneous Traffic

Conditions in Kathmandu using Genetic Algorithm: Case Studies of

Singhadurbar, Padmodaya, and Putalisadak Intersections

by

Aashish Manandhar

A THESIS

SUBMITTED TO THE DEPARTMENT OF CIVIL ENGINEERING

IN PARTIAL FULFILLMENT OF THE REQUIREMENTS FOR THE

DEGREE OF MASTER OF SCIENCE IN TRANSPORTATION

ENGINEERING

DEPARTMENT OF CIVIL ENGINEERING

LALITPUR, NEPAL

DECEMBER, 2023

COPYRIGHT

TRIBHUVAN UNIVERSITY

INSTITUTE OF ENGINEERING

PULCHOWK CAMPUS

DEPARTMENT OF CIVIL ENGINEERING

The undersigned certify that they have read and recommended to the Institute of Engineering for acceptance, a thesis entitled **"Calibration of VISSIM Parameters for Modeling Heterogeneous Traffic Conditions in Kathmandu Using Genetic Algorithm: Case Studies of Singhadurbar, Padmodaya, and Putalisadak Intersections"** submitted by **Aashish Manandhar** in partial fulfillment of the requirements for the degree of Master of Science in Transportation Engineering.

…………………………………………………….

Supervisor: Dr. Rojee Pradhananga

Assistant Professor

Department of Civil Engineering

Institute of Engineering

………………………………………………………

External Examiner: Subhash Dhungel

Traffic and Transportation Engineer, Road Safety

Expert

………………………………………………………

Committee Chairperson: Anil Marsani

Coordinator: M.Sc. in Transportation Engineering

Department of Civil Engineering

Date: ……………………..

ABSTRACT

Most developing countries consists of unsignalized intersections carrying heterogeneous traffic conditions. Microsimulation models and their environment are widely being used for the evaluation of such traffic conditions and development of road networks worldwide. Due to the difficulties in analyzing the complexities of heterogeneous traffic and calibrating the microsimulation, their use in Nepalese context is limited to research studies. This study proposes a procedure to determine the range of values of sensitive calibration parameters tailored to local conditions. Additionally, it suggests a methodology for the automated calibration of microscopic simulation models based on these sensitive parameters at unsignalized intersections with heterogeneous traffic. VISSIM, a microscopic, time step oriented, and behavior-based simulation tool was used to model the intersection which consists of a large number of input parameters making model calibration rather difficult. Sensitivity analysis was performed to identify the sensitive calibration parameters using Latin Hypercube Sampling (LHS) and one way ANOVA testing. Based on the findings of the literature review, 12 calibration parameters were identified. Using traffic flow as a measure of effectiveness, the parameters were reduced to 9 sensitive calibration parameters. The optimum value for these sensitive parameters were obtained by minimizing the error between the simulated and field traffic flow/queue length using genetic algorithm. The calibrated models were then validated based on traffic flow and maximum queue length obtained by performing video graphic survey at the study areas. The simulation models were considered to be validated as the average MAPE values were within the acceptable limits of 15% and GEH values were less than 5. It is expected that the use of these sensitive calibration parameters and their ranges would significantly reduce the time and effort consumed during calibration of the VISSIM models.

Keywords: VISSIM, Simulation, Unsignalized Intersection, Heterogeneous Traffic, Sensitivity Analysis, Automated Calibration, Latin Hypercube Sampling, ANOVA, Genetic Algorithm.

ACKNOWLEDGEMENT

Name: Aashish Manandhar
Roll No.: 078/MSTrE/002

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

LIST OF ABBREVIATIONS AND ACRONYMS

ANOVA	Analysis of Variance
COM	Component Object Model
CORSIM	CORridor SIMulation
DOR	Department of Roads
DOTM	Department of Transport Management
FDOT	Florida Department of Transportation
GA	Genetic Algorithm
GEH	Geoffrey E. Havers
GoN	Government of Nepal
JICA	Japan International Cooperation Agency
LHS	Latin Hypercube Sampling
MAPE	Mean Absolute Percentage Error
MOE	Measure of Effectiveness
MOP	Measure of Performance
OAT	One-At-a-Time
ODOT	Oregon Department of Transportation
PTV AG	Planung Transport Verkehr AG
RMNSE	Root Mean Squared Normalized Error
RMSPE	Root Mean Square Prediction Error
SIDRA	Signalised & Unsignalised Intersection Design and Research Aid
VISSIM	Verkehr In Städten – SIMulationsmodell

CHAPTER 1.   INTRODUCTION

1.1   Background

Traffic microsimulation models and their environment are widely used in the evaluation and development of road networks worldwide. "VISSIM is a microscopic, time step oriented, and behavior-based simulation tool for modeling urban and rural traffic as well as pedestrian flows" (PTV AG 2022). Developing simulation models using such tools requires several steps which include data collection, model formulation, calibration, and validation of the model. Due to the difficulties in analyzing the complexities of heterogeneous traffic and calibrating the microsimulation, their use in Nepalese context is limited to research studies and they are rarely used for assessment of road networks to aid real-life planning and decision making. Thus, the need for a reliable calibration process of the models and parameters is necessary.

VISSIM models are only successful if the model can accurately represent the field conditions and for such accuracy, the model needs to be calibrated. Calibration is the process of fine-tuning the different parameters in the model such that the error between the actual and simulated measures is less than the acceptable value. Calibration and validation are the two most important steps to ensure that the VISSIM models accurately represent real-world conditions. A calibration process is an iterative process that typically involves parameter adjustment to the developed model. Various calibration parameters significantly affect the traffic flow like Wiedemann-74 car following parameters, lane change parameters, lateral behavior parameters, and Wiedemann-99 car following parameters. Calibrating all these parameters is very time-consuming and ineffective for model calibration since all these factors may not affect the model in a significant way depending on local traffic condition. To reduce the computational time and effort, sensitivity analysis is a useful process to identify the sensitive calibration parameters that have a relevant impact on the results of the simulation model.

If calibration is not performed in a model, the outputs generated will be highly erroneous and this process is often carried out with less care as it requires a great deal of time and data. In context of Nepal, calibration in general is performed through manual method using verbal description and trial and error of the default parameters. Manual calibration consumes a lot of time and effort to find the best parameter set as it involves a large combination of parameter sets. In this context, calibration of VISSIM parameters for the local traffic condition in Nepal through parameter optimization becomes vital. Additionally, development of a platform for automated calibration of our intersection models can enhance the accuracy of the analyses and expand their application in real-world planning and management. Parameter optimization can be carried out using various algorithms to obtain optimal values for the parameter sets. Among the various algorithms, Genetic Algorithm (GA) is widely used to solve optimization problems. There is no recorded use of automated calibration currently in our intersections hence proper procedures have not been developed for the automated calibration of the simulation model in our intersections which is observed to be an urgent task.

1.2   Problem Statement

Majority of the Asian countries have heterogeneous traffic conditions which results in a very complex behavior of traffic which is also the case in the majority of our urban intersections. VISSIM software has been highly effective in modeling traffic simulation problems because "simulation is safer, less expensive, and faster than field implementation and testing" (Park and Schneeberger 2003). "Earlier, researchers informally calibrated simulation models and often used default parameters resulting in large errors" (Park and Schneeberger 2003). Most of the earlier studies have focused on homogeneous traffic conditions having good lane discipline. But heterogeneous traffic consists of both motorized and non-motorized vehicles whose static and dynamic characteristics are mixed. Other distinguishing factors include traffic composition where motorcycles are abundant in the context of Nepal as opposed to other countries, side-by-side stacking of vehicles, variable lane widths, and the absence of lane marking and lane discipline across the road. "Motorcycle alone contributed to 75 percent of total volume followed by car with 16 percent during a study performed in New Baneshwor

intersection" (Acharya and Marsani 2020). "In an another study at New Baneshwor intersection, we can see that the majority of modal share is for motorcycles (75.33%), followed by car/van (9.4%), minibus/mini truck (5.91%), pickup/SUV (5.72%), tempo (1.98%), microbus (1.29%), bus (0.21%) and then truck (0.16%)" (Shrestha 2022). "The majority of the fleet in the Kathmandu Valley is made up of motorbikes, which account for 79.1% of the total share. Private vehicles (cars, vans, and jeeps) are next at 12.42%, followed by heavy-duty vehicles at 4%, and public transportation at 2.67%" (Department of Transport Management 2019).

Most of the intersections in Kathmandu are unsignalized resulting in the continuous usage of traffic police for the channelization and effective movement of the traffic which results in unstable and dynamic behavior of the system. Most recently, many major intersections have been signalized with pre-timed traffic signal devices but traffic control is still being done manually by the traffic police officers especially during peak hours. These unsignalized intersections function based on relative priorities of the minor and major roads as seen fit by the traffic police while some are controlled using stop signs on the minor roads.

The analysis of homogeneous traffic conditions with good lane discipline is much easier than analyzing the complexities of heterogeneous traffic. Furthermore, the task of calibrating and validating the microsimulation model for such intersections requires a great deal of time and data. To accurately simulate such systems, the default behavioral parameters should be studied to find out the sensitive behavioral parameters which require modification for calibration and validation of the system model. The informal practice of calibration in our simulation models has wasted a significant amount of time and effort during research. Calibration and identification of range of optimized VISSIM parameters suitable for our local traffic characteristics and the use of automated calibration in VISSIM models to reduce the time and effort of research is negligible in our country. Therefore, a methodology to represent such behaviors in the intersections of Kathmandu is highly necessary. Hence, this study will develop a general procedure for the automated calibration process of microscopic simulation models and suggest a range of parametric values suitable for our traffic condition.

1.3 Objective of Study

The main objective of the study is to establish the range of calibration parameters for VISSIM models applicable to intersections in Kathmandu under heterogeneous traffic conditions, using genetic algorithm. The specific objectives are enlisted as below:

1) To identify the sensitive calibration parameters for VISSIM models.
2) To formulate an automated calibration procedure for determining optimized local calibration parameters tailored to individual intersections.
3) To determine the range of calibration parameters applicable at intersections in Kathmandu.

1.4 Scope of Study

The following scope of the study has been planned:

1) To extract and analyze the data from traffic surveys for the determination of traffic characteristics like volume, directional movement, traffic composition, signal timing, and phase.
2) To collect geometric and vehicle characteristics data of the study areas.
3) To make a VISSIM simulation model based on the collected data of the intersections.
4) To evaluate the model using default parameter values.
5) To identify sensitive parameters based on reviewed literature before performing sensitivity analysis.
6) To perform sensitivity analysis for the selection of the parameters to be used in calibration.
7) To perform automated calibration of the simulation models by fine-tuning the sensitive parameters based on genetic algorithm.
8) To validate the simulation model based on actual field data extracted from video graphic survey and field observations.
9) To determine the local calibration parameters of the individual study area intersections.
10) To determine the range of calibration parameters applicable to intersections in Kathmandu.

1.5   Limitation of Study

The project report was prepared under the following limitations:

1) This study will be performed on a selected number of intersections in Kathmandu.
2) This study uses traffic volume and queue length as Measures of Effectiveness (MOEs) for calibration and validation so further research is recommended using additional MOEs for the study.
3) This study uses only 12 calibration parameters so further research is recommended using additional calibration parameters such as Wiedemann 99 parameters.
4) Vehicle characteristics such as axle configuration and turning radius were not incorporated in the study as it requires extensive amount of data.
5) The study doesn't take into consideration the daily variations and seasonal variations of traffic.

1.6   Organization of Report

The project report consists of the following chapters:

CHAPTER 1: Introduction describes shortly about the practices of calibration of VISSIM models at our urban intersections with heterogeneous traffic conditions. It also pinpoints the problem, objectives, scope, and limitations of the study.

CHAPTER 2: Literature Review discusses the available literature in the practices of sensitivity analysis and calibration process of VISSIM models.

CHAPTER 3: Methodology discusses the various process involved to complete the study.

CHAPTER 4: Results and Discussion includes the analysis of the data collected, sensitivity analysis, automated calibration and validation of the various models using volume and queue length.

CHAPTER 5: Conclusion and Recommendation provides the summary of the results of the research and recommends suggestions for any future works on similar topics.

CHAPTER 2.   LITERATURE REVIEW

2.1   Microscopic Simulation Model

Traffic simulation is useful to accurately recreate traffic as observed and measured in real-time. Microscopic simulation models detail the individual vehicle movements and their interactions within a system and are suitable for the evaluation of complex traffic. Such models become valuable tools to assess the performance of a transportation system because the model can trace the movements of every vehicle. Computer software packages provide the ability to simulate such behaviors and one widely used microsimulation software in Nepal is VISSIM. Consequently, this study focuses on calibrating VISSIM parameters to local traffic conditions in Kathmandu. The following subsections provides more details of the modeling parameters and process to provide further insights of the study.

2.2   VISSIM Software

"VISSIM is a microscopic, time step-oriented, and behavior-based simulation tool for modeling urban and rural traffic as well as pedestrian flows. It can also model rail and road-based public transportation besides private transportation. It is capable of simulating traffic operations on urban streets and freeways, with a special emphasis on public transportation and multimodal transportation. VISSIM uses the psycho-physical perception model developed by Wiedemann (1974) to model driving behavior close to that of the field" (PTV AG 2022). It was developed by PTV Planung Transport Verkehr AG in Karlsruhe, Germany. Traffic microsimulation models and their environment are widely used in the evaluation and development of road networks worldwide but are rarely used in our context. "This may have been caused due to the skepticism which usually results from unrealistic expectations of the capabilities of simulation models and use of poorly calibrated models" (Manjunatha, Vortisch et al. 2013). Due to the difficulties in analyzing the complexities of heterogeneous traffic and calibrating the microsimulation, the need for a reliable, calibrated model is necessary. Figure 2-1 shows the graphical user interface of VISSIM.
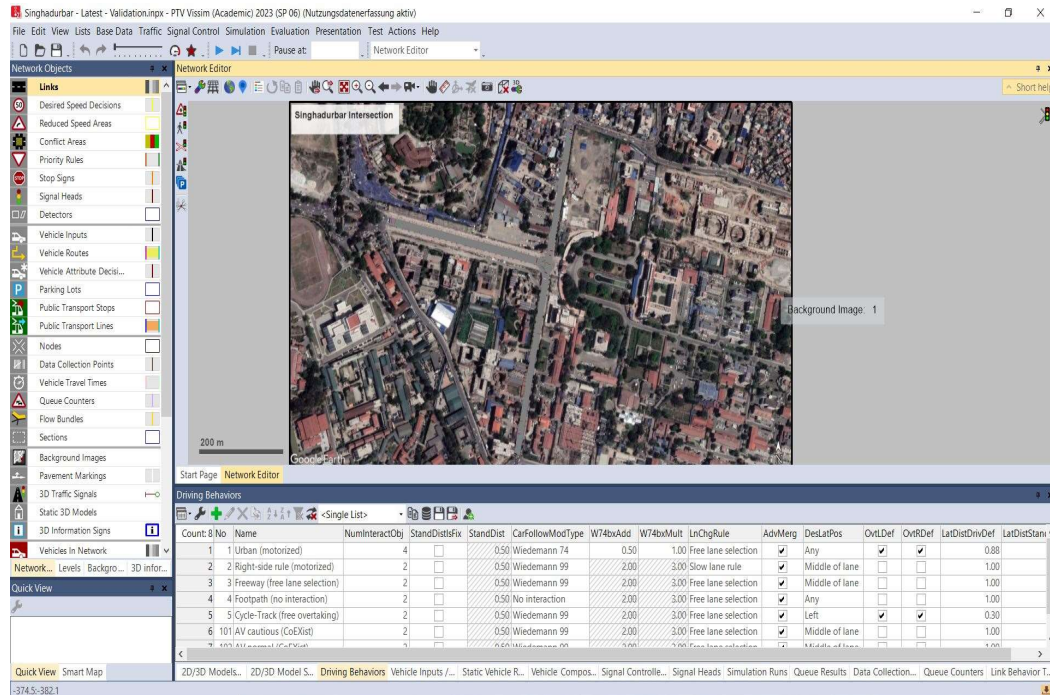
Figure 2-1 VISSIM Graphical User Interface

2.3   Suitability of VISSIM

"Majority of the Asian countries experience mixed heterogeneous traffic conditions, opposed to homogeneous traffic situations with good lane discipline behaviors" (Jayasooriya and Bandara 2018). The majority of the intersections inside Kathmandu are uncontrolled or regulated by traffic police while some of the intersections are signalized. These unsignalized intersections function based on relative priorities of the minor and major roads as seen fit by the traffic police while some are controlled using stop signs on the minor roads. This system can function smoothly if the volume is low but a high volume of vehicles can generate highly complex behavior especially due to the heterogeneous traffic conditions inside Kathmandu.

The traffic composition and behavior characteristics in Nepal are fundamentally different than those in the developed countries where traffic is homogeneous with good lane discipline behaviors. The heterogeneous traffic in our context consists of both motorized and non-motorized vehicles whose static and dynamic characteristics are mixed. The static characteristics include the length, width, etc. of the vehicles while the dynamic ones include speed, acceleration/deceleration, etc. of the vehicles.

While SIDRA's operation is easier, various literatures have shown that the accuracy of VISSIM is higher than SIDRA. Based on vehicle average delay, Tianzi, Shaochen et al. (2013) showed that VISSIM's output was more accurate than SIDRA. Similarly, Montsi (2020) illustrated that the VISSIM model more accurately replicated field performance than the SIDRA model based on the evaluation of performance improvement variations at two signalized intersections.

Sun, Zhang et al. (2013) showed that CORSIM is comparatively more convenient in terms of software usability while VISSIM provides versatile indices for direct output and is found to be more appropriate for beginners. VISSIM is better suited for modeling large intersections with high throughput traffic when considering the intersection average control delay. On the other hand, CORSIM is good at modeling unsaturated intersections. However, neither software package perform well in predicting average queue length, possibly due to large fluctuations in the field situation. Comparably, VISSIM has closer simulation results to the real situation. When evaluating cross-sectional traffic volume, both simulation outputs closely align with field data, with CORSIM showing a slight superiority. This shows that each simulator has its particular advantage in replicating real traffic.

Between the several traffic simulation software, VISSIM was chosen in this study because the study area represents complex operational components such as heterogeneous traffic, abundant motorcycle traffic composition, oversaturated flow, side-by-side stacking of vehicles, variable lane widths, and the absence of lane marking and lane discipline across the road. CORSIM employs a command-line interface, which can be less user-friendly, particularly for individuals not well-versed in command-line operations. In contrast, VISSIM offers a user-friendly graphical user interface (GUI), expanding its accessibility to a broader audience. Furthermore, while CORSIM's graphical visualization features are somewhat limited, VISSIM excels in this regard, presenting intricate visual depictions of traffic flow, including lane-specific animations and 3D modeling. These advanced visualizations can greatly improve comprehension and facilitate effective communication of simulation findings. "VISSIM is also able to model overtaking on same lane, which is not possible in software like CORSIM, AIMSUN and PARAMICS" (Shrestha 2022). However, the simulation needs proper calibration and validation to get a precise result (Mondal and Gupta 2020).

2.4 Driving Behavior and Car Following Model in VISSIM

"VISSIM uses the psycho-physical perception model developed by Wiedemann (1974) to model driving behavior close to that of the field" (PTV AG 2022). Figure 2-2 shows the graphical representation of Wiedemann 1974 car-following model.

Legend:

d = Distance   1 = Free flow state   3 = Approaching state   5 = Collision state
Δv = Change in speed   2 = Following state   4 = Braking state
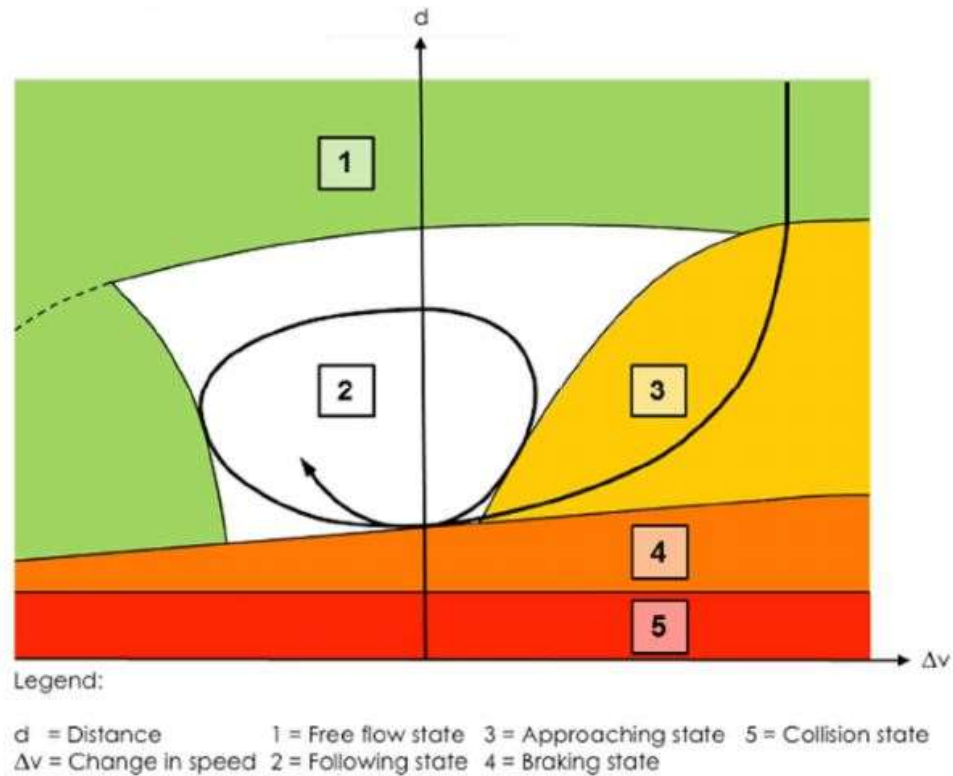
Figure 2-2 Wiedemann 1974 Car following model (Source: PTV AG, 2022)

In the car following model, the primary objective of the following vehicle is to prevent an accident with the leading vehicle while trying to keep up with the desired speed. The figure above shows the classification of the driving process into five main stages, starting with the free flow stage, where the vehicle moves at the desired speed without any interference from nearby vehicles. As the trailing vehicle approaches the leading vehicle and the distance between them decreases, the driver enters the reaction phase, where they observe the leading vehicle and then react accordingly. At this point, the driver of the trailing vehicle realizes that their speed is greater than that of the leading vehicle and enters the perception threshold. The follower starts to decrease their speed upon reaching this threshold, and if the distance between the vehicles is still too small, the vehicle enters the braking state and comes to a stop to maintain a safe distance and prevent a collision state.

2.5   Model Development

The methodology of this study involves the process of site selection, field data collection, determination of the Measures of Effectiveness (MOEs), and the formulation of the simulation model. VISSIM is a powerful tool that offers multiple features to model diverse network systems and has a 3D mode of display that can be used to represent the vehicles, traffic, traffic movements, and the geometry of the study area using the collected field data.

2.6   Sensitivity Analysis

VISSIM consists of multiple parameters that can be adjusted to calibrate and customize the simulation. These parameters can be related to vehicle behavior, traffic control devices, driver behavior, lane-changing behavior, network properties, and other simulation settings. It also includes a range of pre-defined parameter sets that can be used for different types of traffic scenarios. "The users may know the physical meaning of every single parameter in VISSIM, but they will not be able to understand how the simulator calculates the results according to their inputs. Due to the limitation of time and other resources, most calibration procedures cannot afford to calibrate all the parameters in the model. Thus, calibration is carried out only for a limited number of input parameters. However, there is usually no formal procedure for selecting these parameters, other than choosing the ones that appear to the model user as most likely to have a significant effect on the result" (Ge and Menendez 2012).

Sensitivity analysis is an important method used to assess the impact of changes due to the calibration parameters on the model's performance. Sensitivity analysis is a statistical technique that studies the effects of varying one or more variables on a given outcome to identify the most impactful parameters, which inform the calibration process. Multiple methods are available and have been used to conduct sensitivity analysis in VISSIM models which include Pearson correlation coefficient, one-way ANOVA, two-way ANOVA, elementary effects method, quasi-optimized trajectory in elementary effects, Latin Hypercube Sampling (LHS), multiparameter sensitivity analysis, etc.

LHS is a statistical sampling technique that ensures that the sampled values are representative of the full range of parameter values and that they are evenly distributed across the parameter space. So, it reduces the number of samples required to perform computation while covering the entire representative space which makes it a powerful tool for exploring multi-dimensional parameter spaces. Performing sensitivity analysis helps in identifying the most impactful parameters on the model's performance, which informs the calibration process and enhances the accuracy and reliability of the model saving a considerable amount of computational time and effort.

2.7   Calibration and Validation

Calibration and validation are two crucial steps in the development process, as they ensure that the VISSIM model accurately represents real-world traffic conditions. Calibration is the process of fine-tuning the different parameters in the model such that the error between the actual and simulated measures is less than the acceptable value. Calibration typically involves several steps, including field data collection, model creation, adjustment of the sensitive parameters, and iteration. The calibration process is an iterative process, with the parameters being adjusted and the model being re-run until a satisfactory match between the simulated and actual traffic flow data is achieved.

In a survey carried out within the European COST Action TU0903, Punzo (2011) shows that "a great portion of VISSIM users adopt manual trial and error as the standard method for calibration of the model instead of an automatic optimization by computers, a far more efficient and accurate method" as shown in Figure 2-3. The case in our context is even worse with no use of computational techniques for the automatic optimization of calibration parameters of the VISSIM model.

The various techniques used in the optimization of calibration parameters include genetic algorithm, evolutionary algorithm, neural networks, Nelder-Mead (NMA) algorithm, simulated annealing, Quasi-Optimized Trajectories based Elementary Effect (Quasi-OTEE) method, Particle Swarm Optimization (PSO), Downhill Simplex Method, Pareto Archived Dynamically Dimensioned Search (PA-DDS) algorithm, etc. "While various soft computing techniques such as genetic algorithm, neural networks

11

and, tabu search are generally applicable when a large number of parameters needs to be calibrated, manual methods such as trial and error approach are suitable when the number of parameters is small" (Dutta and Ahmed 2019).
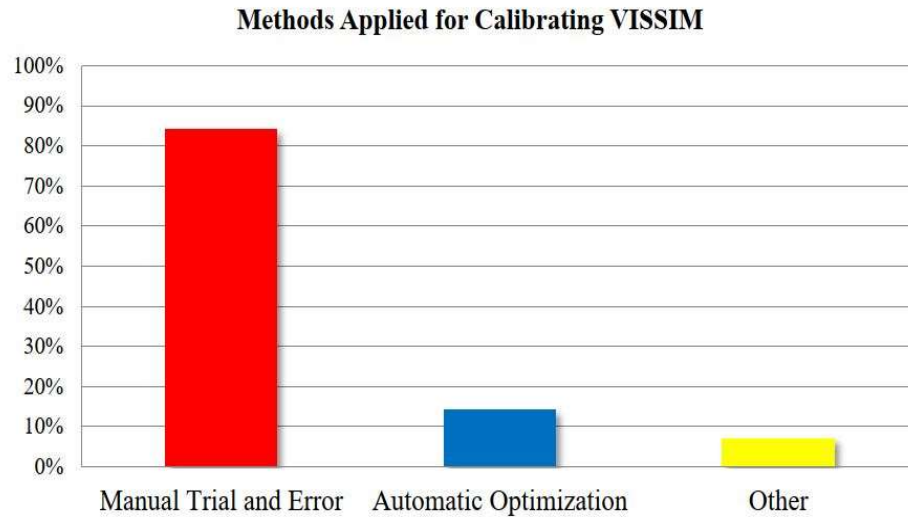
Figure 2-3 Survey results: Methods applied for calibrating VISSIM (Source: Punzo, 2011)

Calibration alone does not guarantee that the model is accurate. Validation of the calibrated model is a critical step to confirm the accuracy and reliability of the model. The goal of validation is to ensure that the calibrated VISSIM model accurately represents real-world traffic conditions under a wide range of scenarios. Validation typically involves several steps, including field data collection, calibration of the formulated model, execution of the model, evaluation and, analysis of the outputs with the field data, adjustment of the sensitive parameters if necessary, and further iteration.

2.8 Genetic Algorithm

Genetic Algorithm (GA) is a heuristic optimization technique based on the concepts of natural selection and natural genetics, a process that operates on chromosomes. It can be used to solve complex optimization problems. "The major steps involved are the generation of a population of solutions, finding the objective function and fitness

function, and the application of genetic operators such as mutation and crossover to produce new solutions" (Pattnaik, Mohan et al. 1998). These new solutions are evaluated and selected based on their fitness, which is a measure of how well they satisfy the optimization criteria.

GA can be used to find the optimal values for the calibration parameters in VISSIM. The calibration parameters become the variables that are adjusted through GA to improve the accuracy of the model. An objective function is defined to express the relationship between the calibration parameters and the performance of the VISSIM model which can be a simple function that measures the difference between the simulated traffic flow and the observed traffic flow, or it can be a more complex function that takes into account multiple performance measures.

Each generation consists of three basic steps: selection, reproduction, and mutation. The GA continues to iterate through generations until a stopping criterion is met. This can be a specified number of generations, or it can be a convergence criterion based on the change in the fitness of the population over time. Once the algorithm has converged, the best solution can be selected as the optimal value for the calibration parameters.

One of the key benefits of using GA for VISSIM calibration is that it can provide a more efficient and effective way to explore the large parameter space of the VISSIM model. Traditional methods such as trial-and-error can be time-consuming and may not find the optimal solution. GA, on the other hand, can quickly search through a large number of potential solutions to find the best fit between the simulated and observed traffic flow. Since this study involves a large parameter space, GA is preferred over the manual trial-and-error method.

A large number of researches have been carried out to develop new techniques of chromosome representation, selection, crossover (recombination), and mutation. Pattnaik, Mohan et al. (1998) used GA in the design of an urban bus transit route network using two coding schemes, namely fixed string length coding and variable string length coding. Park and Qi (2005), Zhizhou, Jian et al. (2005), Mathew and Radhakrishnan (2010), Manjunatha, Vortisch et al. (2013), Siddharth and Ramadurai (2013), Tettamanti, Csikós et al. (2015) are some of the researchers who have

incorporated GA for the optimization of the calibration parameters in their VISSIM models. Rrecaj and MBombol (2015) gave a summary of the differences and similarities of the different practices for VISSIM calibration based on the optimization methodology, the Measure of Performance (MOP) of the fitness function, analysis of parameter consistency, and network types. The study showed that GA was used as an optimization methodology in 6 of the 14 summarized studies. "Genetic algorithm is mostly used and enriched as a heuristic optimization methodology for calibration that underlies on the principle of best individual survival form the population through many iterations" (Rrecaj and MBombol 2015).

The feeding of GA populations can be done using the COM interface of VISSIM which can also send output to the GA as feedback. The VISSIM COM interface allows access to network elements and parameters used in the calibration.

2.9   Sensitivity Analysis and VISSIM Parameter Calibration Practices under Heterogeneous Traffic Conditions

Jayasooriya and Bandara (2018) identified and performed sensitivity analysis on 8 calibration parameters referring to studies on similar countries which resulted in 5 sensitive parameters based on the results of Pearson correlation coefficient and 10% significant level. The model was calibrated at the Katubedda junction and validated at 23 intersections based on queue length by GEH statistics and Mean Absolute Percentage Error. Lownes and Machemehl (2006) used the One-At-a-Time (OAT) method to determine the parameters influencing capacity during congestions.

Mathew and Radhakrishnan (2010) simulated intersections in VISSIM by changing each parameter value by a fixed amount (10%) while keeping the default value for other parameters, and evaluated the sensitivity of the output for each change. This resulted in the identification of 13 sensitive parameters and calibration was performed using the integration of a genetic algorithm in VISSIM through an automation program. The study was performed in 3 similar signalized intersections for non-lane-based mixed traffic conditions in India which used the absolute error in the stopped delay to perform calibration and validation. Cunto and Saccomanno (2008) applied sensitivity analysis

based on variance to identify the sensitive parameters during the calibration and validation of VISSIM in a safety performance study.

Siddharth and Ramadurai (2013) compared the Analysis of Variance (ANOVA) and elementary effects method and found that both methods are effective in finding the sensitive parameters. During the first level sensitivity analysis, the same 5 calibration parameters were found to be sensitive among the 11 parameters chosen for testing from both methods. Second-level sensitivity analysis was performed on the remaining 6 parameters which resulted in further 4 sensitive calibration parameters using the same methods in the first-level sensitivity analysis. The study was performed at a part of an IT corridor and calibration was done using the Visual C++ COM interface of VISSIM and genetic algorithm optimization of sensitive parameters. Comparison of validated flow with the actual field flow to perform calibration and validation using mean absolute percentage error (MAPE).

Dutta and Ahmed (2019) performed sensitivity analysis using the same method as Mathew and Radhakrishnan and one-way ANOVA at three unsignalized three-legged intersections which resulted in the identification of 8 sensitive calibration parameters out of 10 considered parameters for analysis. Calibration of the model was done by minimizing the mean absolute percentage error (MAPE) between the simulated and input traffic flows using the trial and error approach. The model was then validated using input traffic flows and occupancy time as the measure of effectiveness. Manjunatha, Vortisch et al. (2013) performed multi-parameter sensitivity analysis and two-way ANOVA for a check using link capacity as the measure of sensitivity which resulted in the identification of 5 sensitive parameters among the 13 tested parameters. The study was performed at two signalized intersections in Mumbai and calibrated with a genetic algorithm for solving the optimization formulation using field delay as the measure of effectiveness.

In (Park and Qi 2005), ANOVA was used to find the sensitive parameters at an actuated signalized intersection from a set of eight parameters that were to be calibrated. A Latin hypercube experimental design was used to generate 200 sample sets for the eight parameters. The VISSIM model was simulated for the 200 sample sets and one-way ANOVA was performed using average travel time as the measure of effectiveness.

15

Similar to other literature, a genetic algorithm was performed in this study for the optimization of the calibration parameters. The Elementary Effects (EE) method was improved upon by performing a Quasi-optimized trajectory in elementary effects resulting in higher performance and lower computational time for sensitivity analysis. By using this approach in a case study involving a network in the City of Zurich the computation time for sensitivity analysis was greatly reduced (Ge and Menendez 2012).

Rrecaj and MBombol (2015) gave a summary of the differences and similarities of the different practices for VISSIM calibration based on the optimization methodology, the Measure of Performance (MOP) of the fitness function, analysis of parameter consistency, and network types. "Genetic algorithm is mostly used and enriched as a heuristic optimization methodology for calibration that underlies on the principle of best individual survival form the population through many iterations" (Rrecaj and MBombol 2015). The summary showed that different studies considered different measures of performances like volume, travel time, queue length, OD matrix, delay measurement, maximum flow rate, speed-flow, capacity, etc. tested on different statistical tests like F-test, non-parametric Moses, Wilkoxon test, T-test, ANOVA, Chi-square test, Root Mean Square Prediction Error (RMSPE), etc.

In the case of the Nepalese context, Acharya and Marsani (2020) altered 7 calibration parameters at the New Baneshwor intersection simulation model and performed calibration and validation of the model for traffic volume and travel time using GEH statistics and regression analysis. Sharma (2016) used VISSIM microsimulation and recommended the application of U-turn below the Bagmati and Dhobi Khola bridges for the performance improvement of New Baneshwor intersection. Sensitivity analysis were not performed and optimization was done based on a trial and error approach on these studies. Shrestha (2022) performed sensitivity analysis and optimization on 9 selected calibration parameters by varying the values of the parameters adopted from different literature similar to a trial and error approach. The simulation run was conducted by changing one parameter value while keeping the other parameters unchanged from the VISSIM default values. Calibration was done by reducing the cumulative percentage difference between the simulated and actual traffic volume and maximum queue length further validated by GEH statistics and Root Mean Squared

Normalized Error (RMNSE). Table 2-1 and Table 2-2 summarizes the methodology and findings of literatures discussed in this section.

Table 2-1 Summary of the Literature Review

S.N.	Authors of the literature	Study Area	Sensitivity Analysis and Statistical Tests	Number of Calibration Parameters Tested	Number of Sensitive Parameters	Optimization Methodology	Measures of Effectiveness (MOEs)
1	Jayasooriya and Bandara, 2018	Calibration at Katubedda junction. Validation in 23 intersections	Pearson correlation coefficient and 10% significant level	8	5	Best parameter set from trial and error	Queue length by GEH statistics and MAPE (Mean Absolute Percentage Error)
2	Acharya and Marsani, 2020	New Baneshwor signalized intersection	-	7	-	Best parameter set from trial and error	Traffic volume and travel time using GEH statistics and regression analysis
3	Shrestha, 2022	New Baneshwor signalized intersection	Changing one parameter value while keeping the default value for other parameters	9	9	Best parameter set from trial and error	Traffic volume and maximum queue length by GEH statistics and RMNSE
4	Lownes and Machemehl, 2006	Freeway	ANOVA	6 parameter combinations	2 parameter combinations	Simulated Annealing	Capacity
5	Mathew and Radhakrishnan, 2010	3 similar signalized intersections for non-lane-based mixed traffic conditions in India	Changing each parameter value by a fixed amount (10%) while keeping the default value for other parameters	19	13	Genetic Algorithm	Stopped delay
6	Siddharth and Ramadurai, 2013	IT Corridor	Analysis of Variance (ANOVA) and elementary effects method	First Level Sensitivity Analysis – 11, Second Level - 6	First Level Sensitivity Analysis – 5, Second Level – 4	Genetic Algorithm	Traffic flow using MAPE (Mean Absolute Percentage Error)

Table 2-2 Summary of the Literature Review (Contd.)

S.N.	Authors of the literature	Study Area	Sensitivity Analysis and Statistical Tests	Number of Calibration Parameters Tested	Number of Sensitive Parameters	Optimization Methodology	Measures of Effectiveness (MOEs)
7	Dutta and Ahmed, 2019	Three unsignalized three-legged intersections	Changing each parameter value by a fixed amount (10%) while keeping the default value for other parameters and one-way ANOVA	10	8	Trial and error approach	Traffic flow and occupancy time using MAPE (Mean Absolute Percentage Error)
8	Manjunatha, Vortisch et al., 2013	Two signalized intersections in Mumbai	Multi-parameter sensitivity analysis and two-way ANOVA	13	5	Genetic Algorithm	Link capacity and field delay
9	Park and Qi, 2005	Actuated signalized intersection	One-way ANOVA and Latin Hypercube Sampling (LHS)	8	8	Genetic Algorithm	Average travel time
10	Ge and Menendez, 2012	A network in the City of Zurich	Quasi-optimized trajectory in elementary effects	14	5	Only a process of sensitivity analysis	Only a process of sensitivity analysis
11	Zhizhou, Jian et al., 2005	N-S elevated Shanghai Expressway	-	5	-	Genetic Algorithm	Speeds using RMSE (Root Mean Square Error)
12	Tettamanti, Csikós et al., 2015	A test link in Budapest at Oktogon square	-	-	-	Genetic Algorithm	Traffic flow and floating car speed data
13	Pattnaik, Mohan et al., 1998	Urban bus transit route network design	Genetic Algorithm (Description, Procedure, and Parameters)				
14	Rrecaj and MBombol, 2015	Summary of the differences and similarities of the different practices for VISSIM calibration based on the optimization methodology, the measure of performance (MOP) of the fitness function, analysis of parameter consistency, and network types.					

2.10 Findings from Reviewed Pieces of Literature

There have been decades of research dedicated towards microsimulation studies as advancements in computational techniques and automation have flourished. The dynamics of traffic flow especially under heterogeneous traffic conditions is highly complex and its analysis through theoretical models are time consuming and lack accuracy. Microsimulation models such as VISSIM can accurately represent the real world traffic flow dynamics through its flexibility of coding and variability in driving behavior parameters. "As far as the Nepalese context is concerned, numerous research works have been performed in VISSIM" (Shrestha and Marsani 2017, Acharya and Marsani 2020, Shrestha 2022) but they have also implemented the traditional calibration method of trial and error.

Most of the studies involved in calibrating VISSIM have performed sensitivity analysis and automated calibration of VISSIM driving behavior parameters using Analysis of Variance (ANOVA) statistical tests and Genetic Algorithm (GA) optimization methodology respectively under different local settings. These popular methods have been used to model the real world conditions with high accuracy. To the knowledge of the author, studies based on automated calibration of VISSIM parameters for modeling heterogeneous traffic conditions have not been performed in the context of Nepal. Therefore, this study aims to contribute the existing literature by developing an automated calibration procedure and determining the range of values of sensitive calibration parameters applicable at intersections in Kathmandu. The sensitivity analysis involves Latin Hypercube Sampling (LHS) and ANOVA tests, while the automated calibration procedure for VISSIM parameters uses Genetic Algorithm (GA). Both of the procedure employs the use of Python programming language through COM interface of VISSIM to reduce time and effort. The developed procedure will be useful to future researchers to accurately perform manual as well as automated calibration of VISSIM models quickly.

CHAPTER 3.   METHODOLOGY

3.1   Research Design

The research methodology is guided as per the framework shown in Figure 3-1. The study area are selected considering the locations where such studies have not been performed and traffic is regulated manually by the traffic police officers during peak hour.
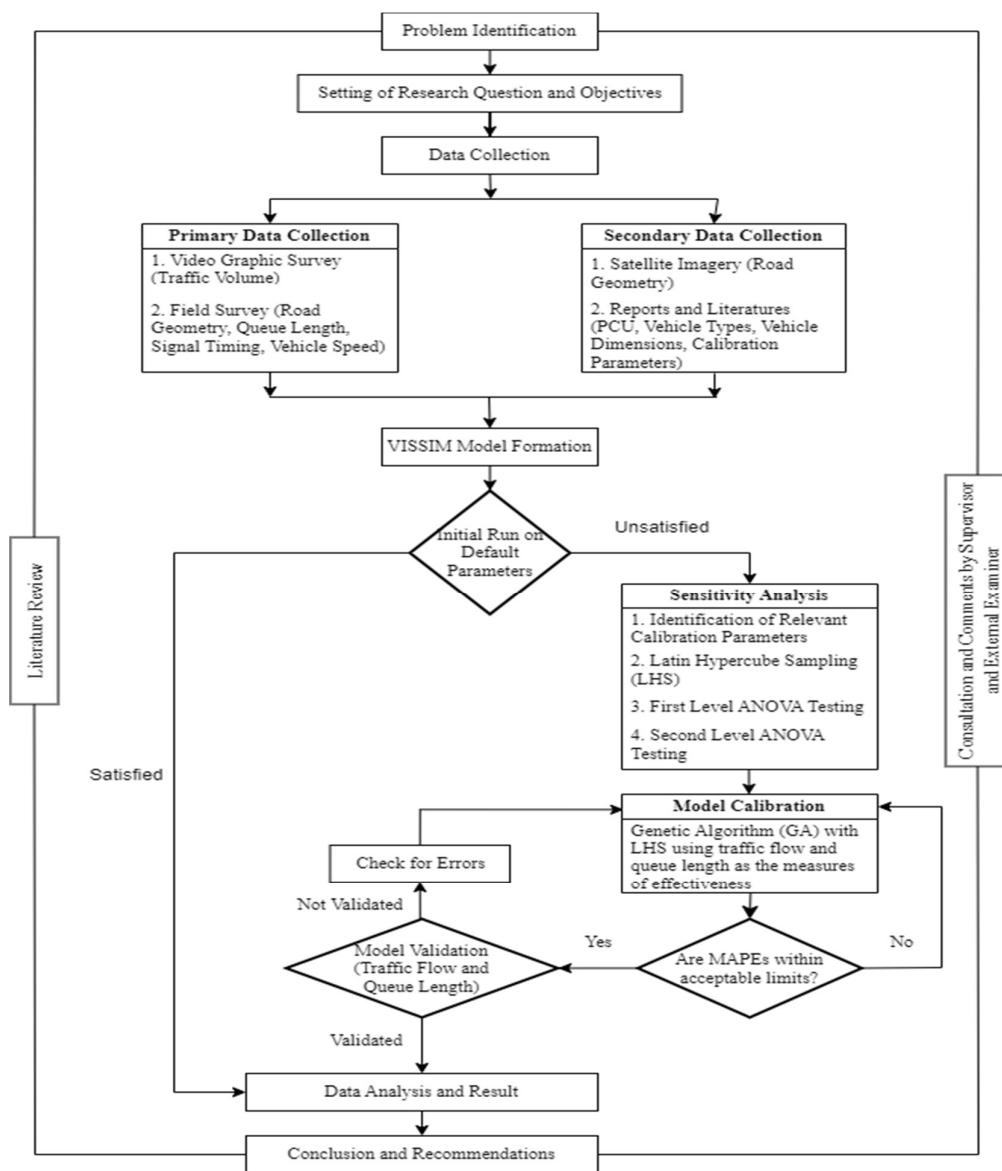
Figure 3-1 Framework of Research

3.2   Study Area

Several pieces of literature have been reviewed regarding the calibration of VISSIM models at intersections. Multiple studies have been done on major intersections of Kathmandu like the New Baneshwor intersection, Balkhu intersection, etc. The study area are selected considering the locations where such studies have not been performed and traffic is regulated manually by the traffic police officers during peak hour. Putalisadak intersection, Padmodaya intersection, and Singhadurbar intersection are found to be suitable for the study. To the knowledge of the author, studies using VISSIM for modeling heterogeneous traffic conditions have not been performed in these intersections. More recently, a study is being carried out by the Road Safety and Traffic Unit, Department of Roads for the implementation of Intelligent Traffic Signal System (ITS) in Kathmandu Valley and these selected intersections are being considered for the signalized improvement under the ITS study.

3.2.1   Putalisadak Intersection

Putalisadak intersection is a major cross intersection lying in the central business district of Kathmandu. Padmodaya followed by Singhadurbar lies to the south, Hattisar to the north, Dillibazar to the east and Bagbazar lies to the west of the cross intersection. The approaches at Hattisar and Padmodaya have a width of 12m with central and lane markings. The east Dillibazar approach has a width of 7m and has central marking. The approach is a one way road with restriction of movement towards Dillibazar from other approaches. This study also considers the east approach as a one way road neglecting the exceptions of traffic moving towards Dillibazar in the data. The west Bagbazar approach also has a width of 7m and has central marking. Pre-timed traffic signal devices have been installed at the intersection but traffic control is done manually by the traffic police officers especially during peak hours. Figure 3-2 and Figure 3-3 shows the general layout and geometry of the Putalisadak intersection.
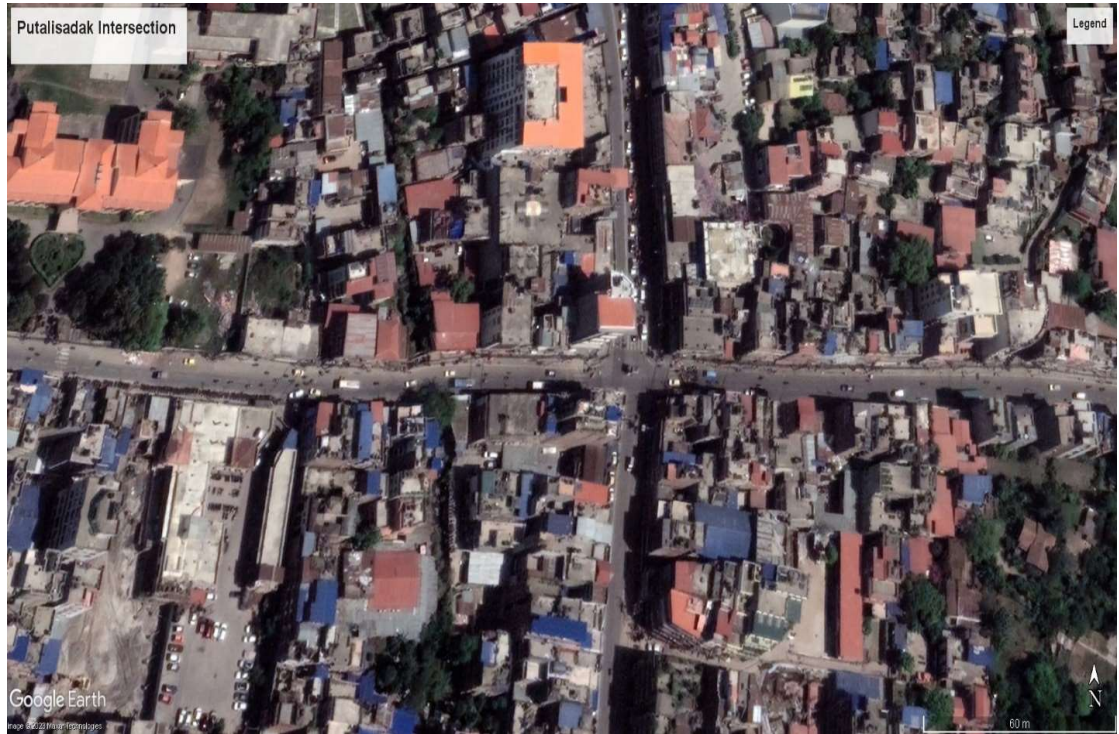
Figure 3-2 Putalisadak Intersection (Source: Google Earth)

24 hour Volume Data			
Flow Direction	Day 1	Day 2	Day 3
Bagbazar - Dillibazar	130	288	163
Bagbazar - Hattisar	9496	11158	10146
Bagbazar - Singhadurbar	78	323	113
Dillibazar - Bagbazar	7451	8660	7959
Dillibazar - Hattisar	7344	8856	8106
Dillibazar - Singhadurbar	9702	11316	10437
Singhadurbar - Dillibazar	22	34	18
Singhadurbar - Bagbazar	6548	7872	7400
Singhadurbar - Hattisar	20783	20695	20870
Hattisar - Dillibazar	36	62	56
Hattisar - Bagbazar	23	34	26
Hattisar - Singhadurbar	25867	30637	27725

Peak hour (10:15 AM - 11:15 AM) Volume Data			
Flow Direction	Day 1	Day 2	Day 3
Bagbazar - Dillibazar	8	16	8
Bagbazar - Hattisar	727	833	645
Bagbazar - Singhadurbar	2	15	3
Dillibazar - Bagbazar	634	743	677
Dillibazar - Hattisar	616	724	667
Dillibazar - Singhadurbar	945	968	1006
Singhadurbar - Dillibazar	0	0	0
Singhadurbar - Bagbazar	729	848	653
Singhadurbar - Hattisar	1645	1735	1811
Hattisar - Dillibazar	5	9	6
Hattisar - Bagbazar	1	2	1
Hattisar - Singhadurbar	2217	2561	2314

Figure 3-3 Geometrical Layout of Putalisadak Intersection (Source: Department of Roads)

3.2.2 Padmodaya Intersection

Padmodaya intersection is a T intersection but it is better represented as a major staggered intersection as there are two T junctions in its vicinity. This study considers the southern T junction which connects Putalisadak to the north, Singhadurbar to the south and Anamnagar to the east of the intersection. The Putalisadak and Singhadurbar approaches have a width of 12m with central and lane markings. The Anamnagar approach has a width of 9m with central marking. Similar to Putalisadak intersection, pre-timed traffic signal devices have been installed at the intersection but traffic control is done manually by the traffic police officers especially during peak hours. Figure 3-4 and Figure 3-5 shows the general layout and geometry of the Padmodaya intersection.

Figure 3-4 Padmodaya Intersection (Source: Google Earth)

PADMODAYA INTERSECTION

24 hour Volume Data

Flow Direction	Day 1	Day 2	Day 3
Singhadurbar - Anamnagar	9341	10173	10153
Singhadurbar - Putalisadak	14651	14722	13143
Putalisadak - Anamnagar	13059	12232	11953
Putalisadak - Singhadurbar	16955	18011	18159
Anamnagar - Singhadurbar	10798	12429	11578
Anamnagar - Putalisadak	14188	15178	15487

Peak hour (10:00 AM - 11:00 AM) Volume Data

Flow Direction	Day 1	Day 2	Day 3
Singhadurbar - Anamnagar	903	849	919
Singhadurbar - Putalisadak	1425	1312	1506
Putalisadak - Anamnagar	937	996	1044
Putalisadak - Singhadurbar	1223	1215	1339
Anamnagar - Singhadurbar	1172	1345	1286
Anamnagar - Putalisadak	1384	1516	1588

Figure 3-5 Geometrical Layout of Padmodaya Intersection (Source: Department of Roads)

3.2.3 Singhadurbar Intersection

Singhadurbar intersection houses important buildings of Nepal Government and is one of the major intersections of Kathmandu. It has a cross configuration connecting Padmodaya followed by Putalisadak to the north, Sahidgate to the west and Maitighar to the south. There is a small central island at this intersection which makes the junction appear like a mini-roundabout but the right turning traffic from Putalisadak are not enforced to comply with the rotary movement through the central island. The Maitighar and Padmodaya approach have a width of 12m with central and lane markings. The west Sahidgate approach also has a width of 12m with central and lane markings but the lane housing the traffic towards Sahidgate only has a width of 3 m. This study excludes the traffic going towards the Singhadurbar access way as it is comparatively low to the other approaches. Similar to the above intersections, pre-timed traffic signal devices have been installed at the intersection but traffic control is done manually by the traffic police officers especially during peak hours. Figure 3-6 and Figure 3-7 shows the general layout and geometry of the Singhadurbar intersection.
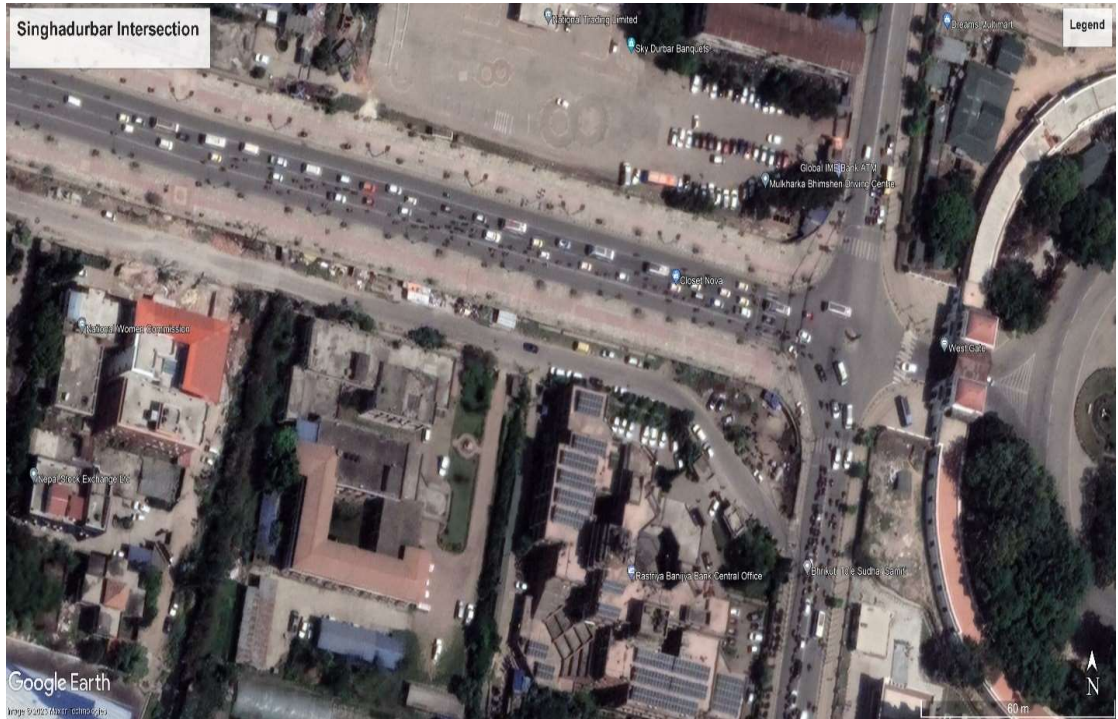
Figure 3-6 Singhadurbar Intersection (Source: Google Earth)

SINGHADURBAR INTERSECTION

24 hour Volume Data

Flow Direction	Day 1	Day 2	Day 3
Maitighar - Sahidgate	5043	5074	5406
Maitighar - Putalisadak	17271	18698	17827
Putalisadak - Sahidgate	4246	4874	4417
Putalisadak - Maitighar	17921	20212	18430
Sahidgate - Putalisadak	5750	6574	5942
Sahidgate - Maitighar	37058	41490	38020

Peak hour (10:00 AM - 11:00 AM) Volume Data

Flow Direction	Day 1	Day 2	Day 3
Maitighar - Sahidgate	740	769	812
Maitighar - Putalisadak	1514	1660	1644
Putalisadak - Sahidgate	361	417	398
Putalisadak - Maitighar	1532	1757	1664
Sahidgate - Putalisadak	338	393	376
Sahidgate - Maitighar	3183	3634	3447

Figure 3-7 Geometrical Layout of Singhadurbar Intersection (Source:

Department of Roads)

3.3 Data Collection and Model Development

3.3.1 Video Graphic Recordings

A microscopic simulation model gives intricate details about the individual vehicle movements and their interactions within the system and such models require an abundant amount of input data. Congestion is relevant in almost all of the intersections within Kathmandu which can often extend beyond one hour peak in heavily congested areas. To analyze such behavior detailed data is required. "A study was carried out by the Department of Roads to collect 72 hours traffic volume count of major 20 intersections inside Kathmandu Valley for signal time design at peak hour volume based on the collected data" (Road Safety and Traffic Unit November, 2021). This study uses the collected data of the 72 hour video graphic recordings of the study area intersections for the model formulation and calibration purposes. A video graphic survey is also carried out for a total duration of two sub intervals (30 minutes) within the peak hour at each intersection for the purpose of model validation.

3.3.2 Data Extraction

VISSIM requires input of following traffic information for model development.
- Vehicle Types
- Classified Vehicle Counts
- Vehicle Compositions
- Directional Movement of Vehicles
- Signal Timing

Different vehicle types are considered in the context of Nepal. "Different types of vehicles take up differing amounts of road space and have different speeds and impose differing loads on the road structure. It is, therefore, necessary to adopt a standard traffic unit to which other types of vehicles may be related. For the geometric design of roads, this standard is the 'Passenger Car Unit (PCU)' which is that of a normal car (passenger car), light van, or pick-up. Other types of vehicles are taken into account by multiplying by the following equivalency factors" (Department of Roads 2013).

Table 3-1 Vehicle Types, Equivalency Factors (Source: Department of Roads, 2013)

S.N.	Vehicle Type	Equivalency Factor
1	Bicycle, Motorcycle	0.5
2	Car, Auto Rickshaw, SUV, Light Van, and Pick Up	1.0
3	Light (Mini) Truck, Tractor, Rickshaw	1.5
4	Truck, Bus, Minibus, Tractor with trailer	3.0
5	Non-motorized carts	6.0

The vehicle types (multi axle truck, tractor, power trailer, non-motorized cart, rickshaw, and auto rickshaw) have not been considered for the vehicle count as they have negligible volume compared to the other vehicle types.

Vietnamese standards recommends the PCU for motorcycles and bicycles as 0.3 and 0.2, respectively for Vietnam, having similar mixed traffic conditions like in Kathmandu. There is no PCU for tempos and microbuses in the Nepal Road Standard. The JICA Survey Team adopted 1.0 as the PCU of tempos, and 1.5 as the PCU of microbuses based on the size of each vehicle (JICA 2012). The PCU values in the Kathmandu Sustainable Urban Transport Project (KSUTP) and Kathmandu Valley Intelligent Traffic Signal Project (KVITSP) reports were adopted as 0.75 for tempos, 1.5 for microbuses, and 0.25 for motorcycles (ADB 2016, Road Safety and Traffic Unit 2023). The PCU values applied by the JICA Survey Team is shown in Table 3-2.

Table 3-2 Passenger Car Unit (Source: JICA, 2012)

Vehicle Type	Bicycle	Motor cycle	Car	Taxi	Light Truck	Tempo	Micro bus	Mini bus	Large Bus	Heavy Truck
Ratio	0.2	0.3	1.0	1.0	1.5	1.0	1.5	3.0	4.0	4.0

Based on the combination of the study reports and Nepal Road Standard, the following PCU values are adopted in this study as shown in Table 3-3.

Table 3-3 Adopted Passenger Car Unit

S.N.	Vehicle Type	Equivalency Factor (PCU)
1	Heavy Truck	3.0
2	Light Truck	1.5
3	Big Bus	3.0
4	Mini Bus	3.0
5	Micro Bus	1.5
6	Car	1.0
7	Motorcycle	0.3
8	Utility Vehicles	1.0
9	4 Wheel Drive	1.0
10	3 Wheeler (Tempo)	1.0
11	Bicycle	0.2

3.3.3 Peak Hour Sample Selection and Data Collection

The input to the microsimulation model should be fed in such a way that the model incorporates all of the traffic congestion present in the study area. The congestion in almost all of the intersections within Kathmandu can often extend beyond one hour peak in heavily congested areas. "Traffic analyses typically focus on the peak-hour traffic volume because it represents the most critical period for operations and has the highest capacity requirements" (Shrestha 2022). The study carried out by the Department of Roads is a large scale study which collects 24 hours traffic volume count for 3 days. The traffic volume was counted at 15 minutes intervals for the different types of vehicles mentioned in Table 3-3. The data is analyzed to identify the peak hour time and traffic volume for this study. So secondary volume data is used for the model formulation of this study.

After the identification of the peak hour period, field data is collected for queue length and signal timing at the study areas during the peak hour. Video graphic survey is also carried out with the help of video cameras at the studied intersections for two sub

intervals (30 minutes) within the peak hour. Traffic data is extracted by manual counting using the video recordings. The survey was conducted at normal weekdays with fair weather conditions on the 21st, 22nd and 23rd of August, 2023. There were no specific disruptions like rally, road crashes or interruptions during the field data collection. There were also no side friction elements such as on-street parking and bus stops present during the field data collection. The volume counts observed in the primary data collection are similar to the secondary data. Therefore, the primary field source data of signal timing and geometric data are used for the model formulation whereas queue length and traffic flow are used for the validation of the model of the study areas.

3.3.4 Development of VISSIM Model

The latest version of PTV VISSIM 2023 (SP 06) "Academic License (PTV AG 2023)" has been used for the development of the model which was provided for six months for this research study. During this research, all simulation work was carried out on a HP Elite Book 830 G8 Notebook PC with an 11th Gen Intel (R) Core i7-1165G7 @ 2.80 GHz processor and 16.00 GB of RAM. The data obtained act as inputs for VISSIM to represent the field conditions. The model development involves the following process.

3.3.4.1 Geometric Data Representation

The first step in the model development involves representing the intersection accurately which is done by using the primary source field observation data and satellite imagery data from Google Earth. The intersection layout includes the number of approaches, width of each approach, length of each approach, turning space and so on. A satellite image is saved using Google Earth Pro and is imported as a background image in VISSIM. The scale is adjusted in the image and the intersection geometry is constructed using links and connectors. The road widths are given as per the field data and the background image is used to ensure that the intersection layout is precisely drawn.

3.3.4.2 Vehicle Representation

Standard vehicle types such as car, bus, truck, motorcycle, bicycles, etc. are available in the software but these models may not perform well under heterogeneous traffic conditions. Nonstandard vehicle types such as tempo and microbus also exist in our intersections. VISSIM has the ability to define vehicle types and change its static and dynamic characteristics. Therefore the next step in the model development involves defining the characteristics of the vehicles in terms of length, width, and speed ranges. Other characteristics include axle configuration and turning radius but require extensive amount of data which is not performed in this study. The adopted size of the different vehicle types used in this study are obtained from websites of vehicle manufacturers (Veerkamp 2015, Indiamart 2023) and different literatures (Mathew and Radhakrishnan 2010, Manjunatha, Vortisch et al. 2013, Dutta and Ahmed 2019, Shrestha 2022) reviewed during the study which are shown in Table 3-4.

Table 3-4 Adopted Average Vehicular Dimensions

S.N.	Vehicle Type	Average Dimensions of Vehicles (m)	
		Length (m)	Width (m)
1	Heavy Truck	8.6	2.5
2	Light Truck	7.5	2.35
3	Big Bus	11	2.5
4	Mini Bus	6.1	2.2
5	Micro Bus	5	1.9
6	Car	3.44	1.45
7	Motorcycle	1.85	0.74
8	Utility Vehicles	4.4	1.5
9	4 Wheel Drive	4.4	1.5
10	3 Wheeler (Tempo)	3.4	1.4
11	Bicycle	1.9	0.45

3.3.4.3   Traffic Representation

The next step involves achieving the actual heterogeneous traffic movement and behavior such as lane changes, overtaking, seepage of smaller vehicles like motorcycles and bicycles to reach the front of a queue, etc. in VISSIM. "The available parameters in the simulation model may not be sufficient to replicate certain special movements by the vehicles in mixed traffic, but depending on the flexibility of the network modeling, one can try to bring the behavior in the simulation as close as possible to reality" (Manjunatha, Vortisch et al. 2013). To emulate the unique behavior in mixed traffic such as aggressive driving, lack of lane discipline, maneuverability of smaller vehicles, etc. the following features can be incorporated in VISSIM.

i.   Vehicle behavior is set to "left hand traffic" regulations.

ii.   General behavior is set to "free lane selection" in lane change driving behavior to allow the vehicles to overtake on each lane.

iii. Desired position at free flow is set to "Any" in lateral driving behavior to place vehicle anywhere within its lane while it is in free traffic flow.

iv.   Diamond queuing takes into account a realistic shape of the vehicles with vehicles positioned offset, such as bikes. Vehicles are internally represented not as a rectangle, but as a rhombus.

v.   Overtaking is allowed from both left and right side in lateral driving behavior. According to traffic rules, right side overtaking of vehicles must be done but in actual field conditions of heterogeneous traffic movement, vehicles approaching a queue uses both left and right side to overtake depending upon the available front gaps.

vi.   The simulation resolution is set to 10 time steps per simulation second similar to (Manjunatha, Vortisch et al. 2013). It specifies how often vehicles and pedestrian move in a simulation second and has an impact on the behavior of vehicles, pedestrians, and the way they interact. "Values between 5 and 10 lead to a more realistic demonstration. This value range is suitable for the production of the final simulation results" (PTV AG 2022). "To adequately model the complex movements in mixed traffic, the calculations have to be at smaller time steps" (Manjunatha, Vortisch et al. 2013). Hence, the simulation resolution is adopted as 10 time steps per simulation second to model the mixed traffic complex movements even though it requires more computational effort.

To further emulate the mixed traffic behavior, multiple driving parameters are selected to be further studied.

3.3.4.4 Vehicle Inputs, Compositions and Vehicle Routing

The data from the study carried out by the Department of Roads also differentiated directional movements at the studied intersections. The volume data is input in terms of vehicle numbers per hour in each approach link of the VISSIM model. The vehicle type classifications and volume are analyzed to get the vehicle compositions which is also input in the VISSIM model. Vehicle proportions of different types of vehicles as per each route is also calculated from the available data and is assigned in VISSIM using static routing decisions. The routes are also fixed in the model from the combination of links and connectors to multiple destinations and are assigned individual vehicle inputs. Peak hour traffic volume data is fed into the simulation model in 15 minutes intervals relative to the simulation period of 1 hour. A warm up period of 5 minutes is provided at the beginning of each simulation run so that the initial empty network is filled with vehicles to allow the simulation model to reach equilibrium.

3.3.4.5 Signal Control

Signal control systems are similar and comparatively easier in homogeneous traffic. Pre-timed traffic signal devices have been installed in all of the studied intersections but traffic control is being done manually by the traffic police officers especially during peak hours. So, primary source of data collection for signal timing during the peak hour is performed at the studied intersections using a stopwatch. The collected signal phase and timing data is divided into 15 minute interval data and classified as red time, amber time and green time. The through and turning movements are also noted during the field observations. Amber time was difficult to pinpoint during the field observations so it was averaged as three seconds preceded by green time. The phase sequence and timings are then input into the signal program and signal heads are inserted at the stop line of each approach.

3.3.4.6 Desired Speed Distribution

The speed of vehicles is obtained by marking a 50 m strip. The time taken by the vehicle to cross the 50 m marked segment is noted in the field. Speed is calculated as the ratio of distance travelled (50 m) by the vehicle to the time taken to travel that distance. 20 samples in each approach leg was performed for random vehicle categories. Since the amount of data samples was small, it was also verified with the speed distribution in similar literatures (Mathew and Radhakrishnan 2010, Dutta and Ahmed 2019, Shrestha 2022). The collected data is found to be in the average speed range provided by (Shrestha 2022). The desired speed range used in this study is as shown in Table 3-5.

Table 3-5 Adopted Minimum and Maximum Speed of Different Vehicle Categories

S.N.	Vehicle Category	Vehicle Types	Average Speed Range (km/h)	
			Minimum	Maximum
1	Two - Wheeler	Motorcycle	15	60
2	Three - Wheeler	Tempo	15	35
3	Four - Wheeler	Car, Micro Bus, Utility Vehicles and 4 Wheel Drive	30	50
4	Buses and Truck	Heavy Truck, Light Truck, Big Bus and Mini Bus	30	45
5	Bicycle	Bicycle	5	15

3.4 Driving Behavior Parameters

VISSIM uses the psycho-physical car following model based on the continued work of Wiedemann and offers different parameter settings to simulate actual field behavior. VISSIM implements two variants of this model, namely Wiedemann-74 and Wiedemann-99 models. The differences in these driving behavior parameters can lead to significant changes on the output of the simulation, so these parameters should be

changed with care. VISSIM models were not originally designed to model heterogeneous and non-lane based traffic as seen in developing countries like Nepal and India. So, different values of different parameters can describe the traffic behavior in different countries with different traffic conditions. Therefore calibration of these parameters is a must to represent the actual field driving conditions. "Wiedemann 74 car following model is used for arterial links while the Wiedemann 99 car following model is used for freeway links" (VDOT 2020). Hence, this study utilizes the Wiedemann-74 car following model which consists of three car following parameters. Based on studied literatures and VISSIM documentation, the parameters that can affect the model are identified below.

   i.   Wiedemann 74 car following model parameters

       a. Average standstill distance (ax): It is the average desired distance between two cars. The tolerance lies within a range of –1.0 m to +1.0 m which is normally distributed at around 0.0 m, with a standard deviation of 0.3 m. The default value of this parameter in VISSIM is 2.0 m.

       b. Additive part of safety distance (bxAdd): It is the value used for the computation of the desired safety distance. The default value of this parameter in VISSIM is 2.0.

       c. Multiplicative part of safety distance (bxMult): It is also another value used for the computation of the desired safety distance. Greater value of this parameter leads to greater distribution (standard deviation) of safety distance. The default value of this parameter in VISSIM is 3.0.

The desired distance d is calculated as given in Equations (3.1) and (3.2) below:

$$d = ax + bx \qquad\qquad (3.1)$$

Where,

ax = standstill distance

$$bx = (bxAdd + bxMult * z) * \sqrt{v} \qquad (3.2)$$

v = vehicle speed (m/s) and z is a value of range [0, 1], which is normally distributed around 0.5 with a standard deviation of 0.15.

Both the additive and multiplicative parts of the desired safety distance determine the saturation flow rate for VISSIM.

ii. Following behavior parameters

    a. Minimum look ahead distance: It is the minimum distance that a vehicle can see forward in order to react to other vehicles either in front or to the side of it (within the same link). It is important when modelling the lateral behavior of vehicles. If several vehicles can overtake within a lane, this value needs to be greater than 0.00 m. For e.g., in urban areas, depending on the speed, the look ahead distance might be approximately 20-30 m, with correspondingly larger values for outside of the city. The default value of this parameter in VISSIM is 0.00 m (PTV AG 2022).

    b. Maximum look ahead distance: It is the maximum distance that a vehicle can see forward in order to react to other vehicles either in front or to the side of it (within the same link). The default value of this parameter in VISSIM is 250.00 m.

    c. Number of interaction objects: It is the number of preceding vehicles and/or number of network objects such as signal heads, reduced speed areas, public transport stops, etc. which the vehicle perceives along its route or path in order to react to them. The default value of this parameter in VISSIM is 4.

    d. Number of interaction vehicles: It is the number of preceding vehicles that the vehicle perceives along its route or path in order to react to them. The number of interaction vehicles is included in the number of interaction objects. The default value of this parameter in VISSIM is 99.

    e. Minimum look back distance: It is the minimum distance that a vehicle can see backwards in order to react to other vehicles behind (within the same link). The minimum look-back distance is important when modeling lateral vehicle behavior. If several vehicles can overtake within a lane, this value needs to be greater than 0.00 m. For e.g., in urban areas it could be 20-30 m, with correspondingly larger values in other places. The default value of this parameter in VISSIM is 0.00 m (PTV AG 2022).

    f. Maximum look back distance: It is the maximum distance that a vehicle can see backwards in order to react to other vehicles behind (within the same link). The default value of this parameter in VISSIM is 150.00 m.

g. Standstill distance for static obstacles: It is the standstill distance (ax) upstream of static obstacles like signal heads, stop signs, PT stops, priority rules, conflict areas. If this option is selected, the vehicles will use the given default value of 0.50 m.

iii. Lane change behavior parameters

a. Waiting time before diffusion: It is the maximum amount of time a vehicle can wait at the emergency stop distance for a necessary change of lanes. When this time is reached the vehicle is removed from the network. The default value of this parameter in VISSIM is 60.00 s.

b. Minimum clearance (front/rear): It is the minimum distance between two vehicles that must be available after a lane change, so that the change can take place. A lane change during normal traffic flow might require a greater minimum distance between vehicles in order to maintain the speed-dependent safety distance. The default value of this parameter in VISSIM is 0.50 m.

c. Safety distance reduction factor: During the lane change, VISSIM reduces the safety distance to the value that results from the multiplication of original safety distance and safety distance reduction factor. The default value of this parameter in VISSIM is 0.6 which reduces the safety distance by 40%. Once a lane change is completed, the original safety distance is taken into account again.

d. Maximum deceleration for cooperative braking: It specifies to what extent the trailing vehicle is braking cooperatively, so as to allow a preceding vehicle to change lanes into its own lane. The higher the value, the stronger the braking and greater the probability of changing lanes. The default value of this parameter in VISSIM is -3.00 m/s^2.

iv. Lateral behavior parameters: "Apart from car following parameters, lateral distances of various vehicle types contribute to the VISSIM output, as a little variation in these can result in a varied output in simulation. The seepage of two-wheelers can be simulated in VISSIM by giving appropriate lateral distances to be maintained at rest and 50 km/h respectively and setting the parameters collision time gain, minimum longitudinal speed for lateral movement and time between direction changes" (Manjunatha, Vortisch et al. 2013).

a. Collision time gain: It is the minimum value of the collision time gain for the next vehicle or signal head, which must be reached so that a change of the lateral position on the lane is worthwhile and will be performed. The collision time is calculated based on the desired speed of the vehicle. The default value of this parameter in VISSIM is 2.00 s.

b. Minimum longitudinal speed: It is the speed which still allows for lateral movements. The default value of this parameter in VISSIM is 1 km/h which ensures that vehicles can also move laterally if they have almost come to a halt already.

c. Time between direction changes: It defines the minimum simulation time which must pass between the start of a lateral movement in one direction and the start of a lateral movement in the reverse direction. The higher this value, the smaller are the lateral movements of vehicles. These lateral movements only take place if overtaking on the same lane is permitted. Lateral movement for a lane change is not affected by this parameter. The default value of this parameter in VISSIM is 0.00 s.

d. Minimum lateral distance: It is the minimum distance between vehicles when overtaking within the lane and keeping the distance to vehicles in the adjacent lanes. It is further divided into:

- Distance standing at 0 km/h: The default value of this parameter in VISSIM is 0.20 m.

- Distance driving at 50 km/h: The default value of this parameter in VISSIM is 1.00 m.

The minimum distance is linearly interpolated for other speeds than at 0 km/h and 50 km/h.

3.5 Initial Evaluation

Once the simulation model is set up, the model is run based on the default parameter settings and the output are compared to the input data. If a close match is found between them, the model is deemed appropriate for further analysis but this may not be a case in general. "A minimum 10 runs, using different random seeds for each run are required to adequately account for variations in network operational performance" (VDOT

2020). In this study, 5 simulation runs each with different random seed value is performed similar to the studied literatures (Park and Qi 2005, Park, Won et al. 2006, Siddharth and Ramadurai 2013).

3.6 Sensitivity Analysis

Sensitivity analysis is a statistical technique that studies the effects of varying various parameter assumptions to the outcome of the process. VISSIM provides multiple parameters some of which are explained above that can be adjusted to calibrate and influence the simulation. "In VISSIM microscopic simulation software, there are around 40 parameters which can be changed for modeling the driver behavior patterns" (Jayasooriya and Bandara 2018). Performing analysis on all the parameters offered by VISSIM will consume a considerable amount of time and effort. "A proper sensitivity analysis, including the initial screening of the parameters, can be very valuable for the subsequent calibration process. Moreover, it may actually reduce the total efforts needed during the model calibration. A good sensitivity analysis could provide both quantitative and qualitative information regarding the effects of the different model parameters (and their variations) on the simulation results" (Ge and Menendez 2012). Hence, sensitivity analysis is performed to identify the relevant and sensitive calibration parameters for the study.

The sensitivity analysis involves the process of incrementing the value of the identified calibration parameters in small units and analyzing the effect on the simulation output. The process involves a large number of simulation runs taking up a considerable amount of time. Multiple simulation runs are performed with different random seeds to reduce the effect of stochasticity. The steps involved are shown below.

3.6.1 Identification of Relevant Calibration Parameters

Based on studied literatures on similar studies in Kathmandu and other Asian countries with heterogeneous traffic conditions and personal engineering judgment, 12 driving behavior parameters are considered important and selected for sensitivity analysis. "Each parameter needs a lower and upper bound the value can take so that the

optimization model needs to search in lesser space which makes the procedure computationally efficient. Such a range is necessary to ensure realistic performance of the simulation model" (Mathew and Radhakrishnan 2010). The acceptable range for these parameters are also fixed based on the data from studied literatures (Park and Qi 2005, Park, Won et al. 2006, Mathew and Radhakrishnan 2010, Manjunatha, Vortisch et al. 2013, Siddharth and Ramadurai 2013, Jayasooriya and Bandara 2018, Dutta and Ahmed 2019, Acharya and Marsani 2020, Shrestha 2022) including the VISSIM user manual (PTV AG 2022). The studied parameters, their default values and the adopted ranges in the study are shown in Table 3-6.

Table 3-6 Studied Parameter List and their Range of Values

S.N.	Driving Behavior	Parameters	Unit	Default value	Range of values based on studied literatures
1	Wiedemann- 74 Car Following Parameters	Average Standstill Distance	meter	2	[0.3 – 2]
2		Additive Part of Safety Distance	-	2	[0.1 – 2]
3		Multiplicative Part of Safety Distance	-	3	[0 – 3]
4	Following Behavior Parameters	Minimum Look Ahead Distance	meter	0	[10 – 30]
5		Maximum Look Ahead Distance	meter	250	[200 – 350]
6		Minimum Look Back Distance	meter	0	[5 – 30]
7		Maximum Look Back Distance	meter	150	[80 – 180]
8	Lane Change Behavior Parameters	Waiting Time Before Diffusion	second	60	[30 – 75]
9		Minimum Clearance (Front/Rear)	meter	0.5	[0.1 – 1]
10		Safety Distance Reduction Factor	-	0.6	[0.2 - 0.7]
11	Lateral Behavior Parameters	Minimum Lateral Distance (Standing) at 0 km/h	meter	0.2	[0.1 - 0.5]
12		Minimum Lateral Distance (Driving) at 50 km/h	meter	1	[0.6 – 1]

3.6.2 Latin Hypercube Sampling (LHS)

Latin Hypercube Sampling (LHS) is a powerful and widely-used statistical technique for sampling from multi-dimensional parameter spaces in a systematic and efficient manner. Unlike traditional random sampling methods, LHS ensures a more representative and evenly spaced coverage of the input variables, making it particularly valuable when dealing with complex systems and computationally expensive simulations or experiments (McKay, Beckman et al. 2000). The fundamental idea behind Latin Hypercube Sampling is to create a stratified and space filling sampling design that ensures even and representative coverage of the input parameter space while reducing sampling variance. Unlike traditional random sampling, where each parameter is randomly chosen without any constraints, LHS divides each parameter into equally spaced intervals and then selects one sample per interval for each parameter. The resulting samples form a "Latin hypercube," which refers to the Latin square-like structure of the data points.

The primary advantages of Latin Hypercube Sampling are as follows.
- Improved space-filling properties and better coverage of the entire parameter space.
- Reduced correlation between samples i.e. even distribution of the samples across each parameter's range.
- Efficient exploration of high-dimensional spaces.
- Better representation of uncertainty allowing a more robust analysis.

The steps involved in Latin Hypercube Sampling are as follows.
- Define the space, ranges and constraints for each input parameter.
- Divide each parameter range into equally spaced intervals.
- Randomly permute the samples within each interval.
- Combine the samples from all parameters to create the final Latin hypercube sample set which ensures that no two samples share the same row or column.

After the identification of relevant calibration parameters, the number of combinations for testing among these parameters will be extremely large. The amount of time to

consider all these possibilities will be extremely large. For e.g., if 5 values are provided to each studied calibration parameters, it would generate $5^{12} = 244140625$ possible combinations. The amount of time to conduct simulation runs for each possible combination along with multiple runs with different random seeds to reduce stochastic variability would have been immense. Based on similar literatures, 200 scenarios are deemed to be adequate to cover the entire parameter surface. Hence, in this study 12 calibration parameters each with three to five different values inside the range defined in Table 3-6 are used to generate 200 scenarios. 5 random seeded simulation runs of the 200 scenarios are performed in VISSIM, for a total of 1000 runs. The 5 random seeded runs are then averaged to represent the results of each of the 200 parameter sets.

Using LHS, an orthogonal array that randomly samples the entire design space broken into regions of equal probability is created. 200 sets of 12 parameters is created using LHS and each set of parameter values is simulated for 5 random seeds. The VISSIM model for the intersections under study are simulated for all these random sets and the error between the actual and simulated traffic volume is collected. VISSIM has a COM interface which can be accessed to call and simulate VISSIM externally through a code. To reduce time and effort, a programming code is written in Python programming language through COM interface of VISSIM to create parameter sets from LHS, run the simulation, and collect the output in a separate Excel file. The randomly created samples from LHS are then made discrete, grouped and indexed. The programming code used in the first level sensitivity analysis is provided in APPENDIX E.

3.6.3   First Level ANOVA Testing

Analysis of variance is being widely employed for obtaining the optimal set of parameters (Park and Qi 2005, Lownes and Machemehl 2006). Siddharth and Ramadurai (2013) have shown the effectiveness of using ANOVA in finding the parameters that are sensitive in a significant way. "ANOVA tests the null hypothesis that the means for several groups in the population are equal by comparing the sample variance estimated from the group means with that estimated within the groups" (Arnold and Milton 2003). SPSS, a statistical package is used for this analysis. It is used to draw conclusion about population means when the means are affected by different

factors and shows whether the particular parameter affects the output of the simulation. This procedure can reduce the number of the several independent parameters to those factors which significantly affect the response variable which saves a significant amount of time and effort in simulation.

The discrete values of the parameter from the LHS samples and the change in error with respect to the default parameter set values are input into SPSS to perform one way ANOVA testing. The change in error is calculated by comparing the traffic volume obtained using default parameter values with those obtained by changing the parameter values given in Table 3-6. Trials are carried out on the different approach legs of the intersections under study. When the significance value of the F-test (p-value) is smaller than the user defined confidence level, the null hypothesis is rejected, thereby indicating that the group means are statistically different. Following similar literatures (Siddharth and Ramadurai 2013, Dutta and Ahmed 2019), the parameters with small p-values less than 0.2 are identified as sensitive parameters in this study.

3.6.4   Second Level ANOVA Testing

After performing first level sensitivity analysis, it may be determined that some of the parameters are not sensitive, prompting the performance of a second level sensitivity analysis. The reason for doing so is the possibility of highly significant parameters dominating the effect of marginally significant parameters. Similar to the first level testing, the same code with some minor adjustments is run by removing the sensitive parameters identified from the first level of testing. 200 sets of those parameters which are not identified as sensitive during the first level testing is created using LHS and each set of parameter values is simulated for 5 random seeds again for a total of 1000 runs. The VISSIM model for the intersections under study are simulated for all these random sets and the error between the actual and simulated traffic volume is collected. The output is collected in a separate Excel file implemented by the code. The randomly created samples from LHS are then made discrete, grouped and indexed. The discrete values of the parameter from the LHS samples and the change in error from default parameter set values are input into SPSS to perform one way ANOVA testing again. Trials are carried out on the different approach legs of the intersections under study.

Similarly, the parameters with small p-values less than 0.2 are identified as sensitive parameters. The programming code used in the second level sensitivity analysis is provided in APPENDIX F.

3.7 Calibration

Calibration is the process of fine-tuning the different parameters in the model such that the error between the actual and simulated measures is less than the acceptable value. Once the sensitive parameters have been identified from the sensitivity analysis, they need to be adjusted to accurately represent the field conditions in the model, while keeping the other parameter values constant. The calibration of the model is done using the peak hour traffic data from the first and second day of the three-day traffic count data at each intersection. The queue length is also being simultaneously checked during the calibration process. There are majorly two methods of calibration which are listed below.

i. **Trial and Error method (Manual calibration):** A conventional optimization technique that is time-consuming when the traffic to be represented is complex.

ii. **Automated calibration:** A large amount of time and effort is required to find the best parameter set from the large number of combination of the parameter values. "Therefore it is better that the process of calibration be automated to search for the best parameter set" (Siddharth and Ramadurai 2013). The various techniques used in automated calibration include genetic algorithm, evolutionary algorithm, neural networks, Nelder-Mead (NMA) algorithm, simulated annealing, Quasi-Optimized Trajectories based Elementary Effect (Quasi-OTEE) method, Particle Swarm Optimization (PSO), Downhill Simplex Method, Pareto Archived Dynamically Dimensioned Search (PA-DDS) algorithm, etc.

The literature review identifies genetic algorithm (GA) as a popular optimization methodology used by multiple researches. "Many of the previous works have used optimization with GA as a tool for calibration due to its robustness and ability to handle a large number of parameters" (Mathew and Radhakrishnan 2010). So, this study also utilizes genetic algorithm which is a heuristic optimization technique to perform

automated calibration of the VISSIM model. VISSIM has a COM interface which can be accessed to call and simulate VISSIM externally through a code. A programming code is written in Python programming language with the help of a Genetic Algorithm package through COM interface of VISSIM. The code creates parameter sets from LHS, loads the network, runs the simulation, performs automated calibration, checks the fitness function, and compiles the output in a separate Excel file. The steps involved in the automated calibration process for this study are shown in Figure 3-8 below.
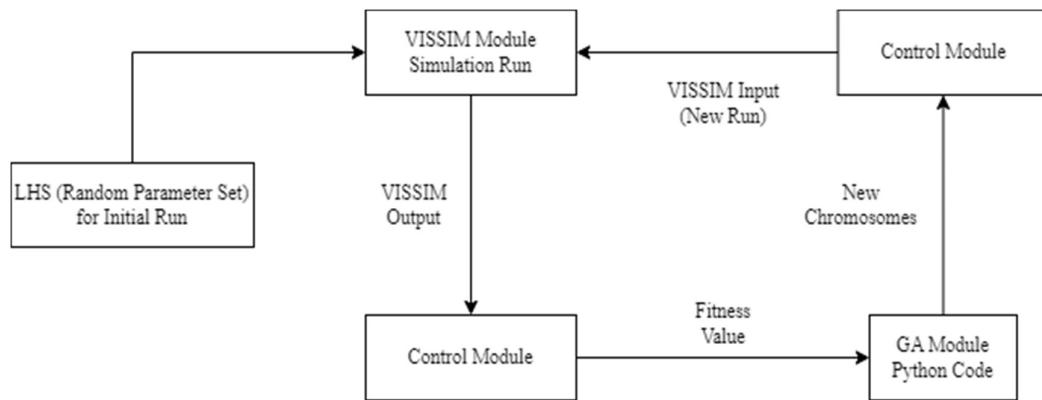
Figure 3-8 Program Modules for VISSIM Calibration

3.7.1 Measure of Effectiveness (MOE)

Measure of effectiveness is a single or multiple factors which provides a basis for evaluating the performance of a transportation system. The choice of an effective measure influences the calibration process. "Traffic volume has been widely implemented by various researchers as a basic MOE for calibration and validation in signalized intersections (Siddharth and Ramadurai 2013, Acharya and Marsani 2020, Shrestha 2022), roundabouts (Arroju, Gaddam et al. 2015), and unsignalized intersections (Caliendo 2014, Dutta and Ahmed 2019)". In this study, traffic volume and queue length have been selected as the key measure of effectiveness for calibration and validation. The queue length for the secondary data source was not available during the study. The volume counts observed in the primary data collection are similar to the secondary data and the traffic conditions during those periods are somewhat similar in signal timing, vehicle routing, geometric characteristics and seasonal conditions.

Hence, the queue lengths obtained in the field are used in the automated calibration process.

3.7.2 Statistical Checks for Calibration

The simulation models are initially run with default parameters. The generated output is used to find the initial error between the simulation output and observed measurements using MAPE. The mean absolute percentage error (MAPE) is a measure of prediction accuracy of a forecasting method in statistics. "Mean Absolute Percentage Error (MAPE) has been widely used for testing the goodness of fit by various researchers" (Siddharth and Ramadurai 2013, Arroju, Gaddam et al. 2015, Jayasooriya and Bandara 2018, Dutta and Ahmed 2019). It usually expresses the accuracy as a ratio defined by the formula as given in Equation (3.3) below.

$$MAPE = \frac{1}{N} * \sum_{i=1}^{N} E\text{i} \tag{3.3}$$

Where,

E_i: Absolute percentage error at interval 'i' as given in Equation (3.4).

$$E_i = \frac{|Actual\ Qi - Simulated\ Qi|}{Actual\ Qi} * 100\% \tag{3.4}$$

Actual Q_i: Actual stream flow at interval 'i'

Simulated Q_i: Simulated stream flow at interval 'i'

Different literatures have recommended and accepted the threshold value for MAPE as a statistical check. Dowling, Skabardonis et al. (2004), Mathew and Radhakrishnan (2010), Manjunatha, Vortisch et al. (2013), Dutta and Ahmed (2019) adjusted the sensitive parameters to bring down the errors to an acceptable limit of 15%.

The best universal measure to compare traffic flow is the Geoffrey E. Havers (GEH) formula which has been used by numerous researchers (Jayasooriya and Bandara 2018, Acharya and Marsani 2020, Shrestha 2022). GEH statistic is a modified chi-square statistic that accounts for both absolute and relative errors defined by the formula as given in Equation (3.5) below.

$$GEH = \sqrt{\frac{2\,(M-C)^2}{(M+C)}} \qquad (3.5)$$

Where,

M: Model traffic volume in vehicles per hour

C: Input traffic volume in vehicles per hour

Different literatures have recommended calibration and validation threshold values. The values recommended by the Oregon Department of Transportation (ODOT) are shown below in Table 3-7.

Table 3-7 GEH Statistic Guidelines (Source: (ODOT 2011))

GEH < 5.0	Acceptable Fit
5.0 < = GEH < = 10.0	Caution: possible model error or bad data
GEH > 10.0	Unacceptable

Similarly, the threshold values recommended by the Florida Department of Transportation (FDOT) are shown below in Table 3-8. In this study, MAPE threshold value of 15% and GEH threshold value of 5.0 are used.

Table 3-8 Classical Model Calibration Targets (Source: (FDOT 2014))

Calibration Item	Calibration Target/Goal
Traffic Volume	Simulated and measured link volumes for more than 85% of links to have a GEH statistic value of five (5) or lower.
	Sum of link volumes to have a GEH statistic value of five (5) or lower.
Queue Length	Difference between simulated and observed queue lengths to be within 20%.

3.7.3   Genetic Algorithm with Latin Hypercube Sampling (LHS)

"Genetic Algorithms (GAs) are powerful and versatile optimization techniques inspired by the process of natural selection and evolution in biology. Developed by John Holland

in the 1970s, GAs are a part of the broader field of evolutionary computation and have proven to be highly effective for solving complex optimization and search problems in various domains. The fundamental concept behind Genetic Algorithms lies in simulating the process of natural selection to evolve a population of potential solutions over successive generations. The algorithm mimics the principles of natural selection, including reproduction, mutation, and survival of the fittest, to find an optimal or near-optimal solution to a given problem" (Holland 1992). Genetic Algorithms have become a valuable and effective tool for finding near-optimal solutions for complex optimization problems. "However GA do not necessarily guarantee global optimality in the mathematical sense" (Ma and Abdulhai 2002). The methodology for the use of genetic algorithms is guided as per the framework shown in Figure 3-9 below.

Chromosome Representation

Random Initialization of the Population using LHS

Fitness Function Evaluation = Inverse of MAPE

Generation of the Next Population (Genetic Operation)

Selection

Completion of a cycle/generation

Crossover

Mutation

No

Replacement (Reinsertion)

Stopping Criteria (Maximum number of generations = 20)

Yes

Optimal Solution
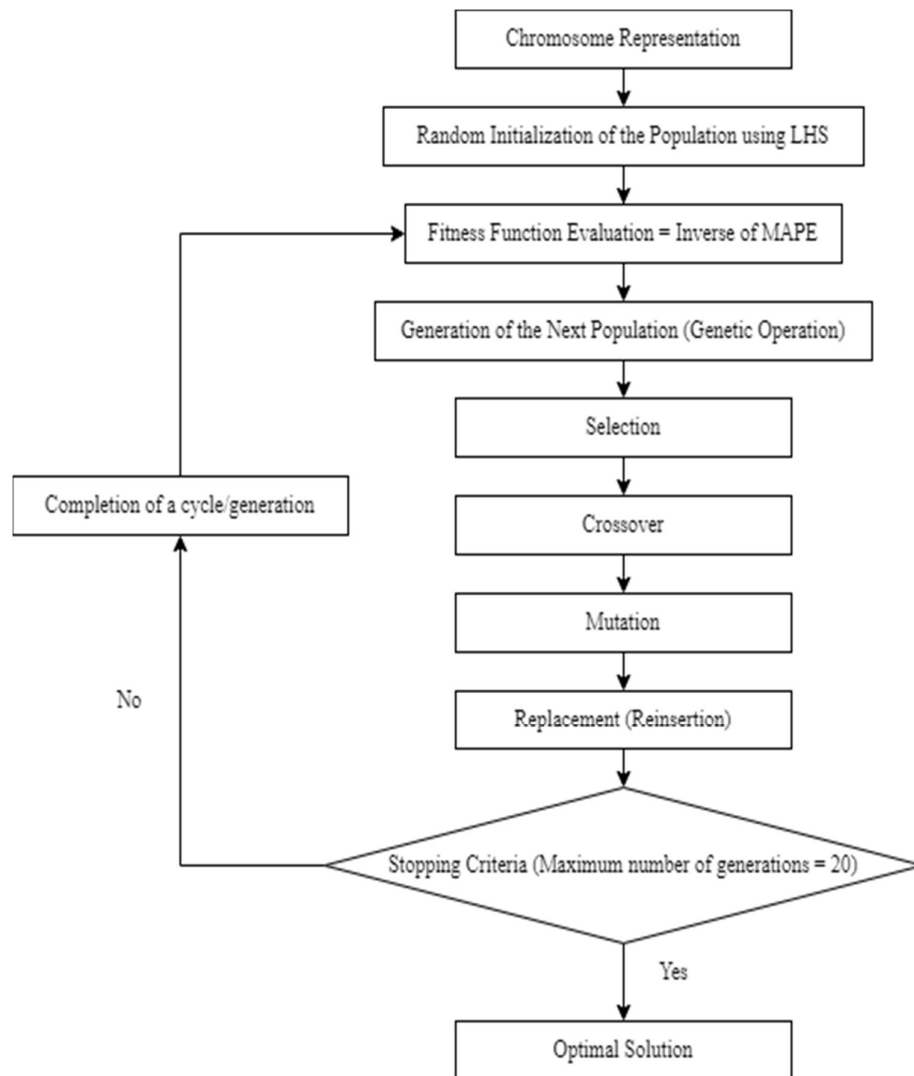
Figure 3-9 Methodological Framework of Genetic Algorithm

"The GA solution to a problem generally proceeds by completing the six basic steps of chromosome representation, population initialization, objective value evaluation, selection, genetic operators and reinsertion" (Pradhananga, Hanaoka et al. 2011). The sensitive parameters are optimized based on an optimization problem which minimizes the MAPE as shown in Equations (3.6) and (3.7) below between the observed and simulated flow/queue length for each intersection as given in Equation (3.4) above. The optimization problem in context to this study can be written as:

Minimize

$$MAPE = \frac{1}{N} * \sum_{i=1}^{N} E_i$$

(3.6)

Subjected to,

$$\beta^j_{min} \leq \beta^j \leq \beta^j_{max}, \qquad \forall\, j$$

(3.7)

Where,

E_i: Absolute percentage error at interval 'i' as given in Equation (3.4).

Actual Q_i: Actual stream flow at interval 'i'

Simulated Q_i: Simulated stream flow at interval 'i'

β^j_{min}:   Lower range of the jth parameter

β^j: Value of the jth parameter

β^j_{max}:   Upper range of the jth parameter

After the identification of the sensitive parameters and their acceptable ranges, a GA is applied to find the optimal parameter values. GA is a heuristic optimization technique based on the concepts of natural selection and natural evolution, a process that operates on chromosomes. It is a must that VISSIM and GA are integrated to run simultaneously at every iteration. A programming code is written in Python programming language with the help of a Genetic Algorithm package through COM interface of VISSIM. The GA begins by randomly initializing a population of initial chromosomes in which each chromosome represents a set of combinatorial parameters (genes). The chromosomes are passed through the COM interface to the VISSIM model and a simulation run is then automatically triggered based on that configuration. After the end of the simulation, VISSIM returns the generated results (feedback) back to GA, which in turn evaluates the fitness function value. The total number of simulation runs is determined by the population size and the total number of new chromosomes produced through all

generations. On the basis of the fitness function value of each chromosome in the population, the GA conducts genetic operations which involves the process of selection, crossover, mutation, and replacement to produce a new set of chromosomes. The GA continues for multiple iterations until a stopping criteria is met which can be convergence, saturation of the fitness function value, or by reaching a specified maximum number of generations to find the least MAPE value between the observed and simulated measures. The programming code used in the automated calibration process is provided in APPENDIX G. The program modules for the code is shown in Figure 3-8 above. The steps involved in the genetic algorithm process are as follows:

i. **Chromosome representation:** Genetic algorithm require an appropriate representation of the potential solutions to the problem. A chromosome can be defined as a set of values of combinatorial parameters for the simulation model. A chromosome is subdivided into genes, which is a binary representation of a single parameter value with a specified lower and upper bound. In this study, the genes are represented by the identified sensitive parameters with their value ranges as shown in Table 3-6. "The length of the bit string is of paramount importance. It determines how precise a point the GA could reach in the search space. The longer the binary bit string is, the better, but this comes at the expense of high computation cost" (Ma and Abdulhai 2002). For better results and precision, the gene type is chosen as float string. The representation of gene, chromosome and population is shown in Figure 3-10 below.
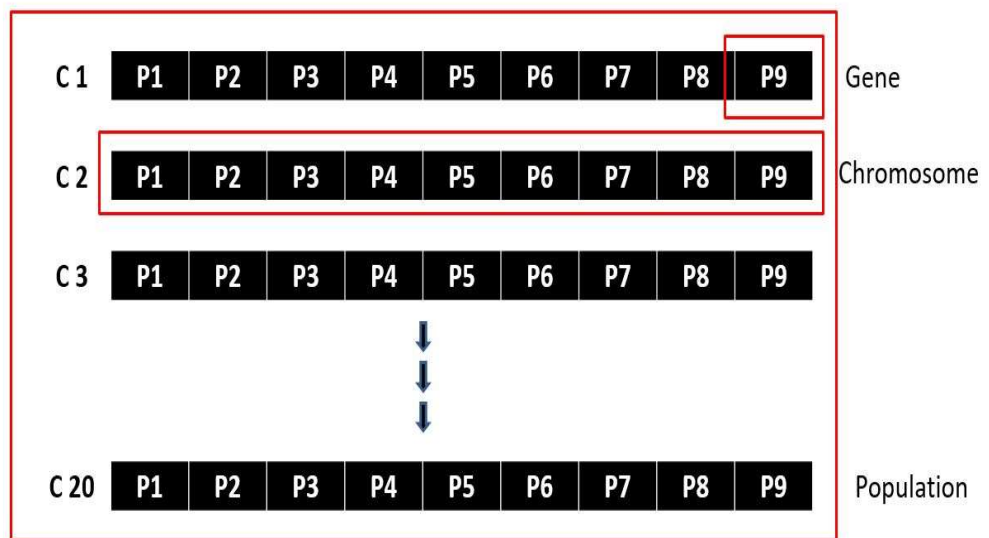
Figure 3-10 Gene, Chromosome and Population Representation

ii. **Random initialization of the population:** During the simulation run of the represented chromosomes, a population of individuals is created, with each individual representing a potential solution. GA operates on these several potential solutions. The initial population data is randomly generated using Latin Hypercube Sampling (LHS), which gives an unbiased output and utilizes a wide range of data. Based on various literatures, the population size is chosen as 20 among which half of the population are allowed to mate.

iii. **Fitness function evaluation:** The fitness function value is the measure of goodness of an individual solution in the population to solving the optimization problem. Higher fitness value candidates have a higher probability of being selected for further examination. The simulation outputs are compared to the real target values to find the deviations from the target. Smaller deviations inch towards better solutions. Traffic volume is used in the evaluation of the fitness function as the inverse of MAPE values while simultaneously checking the MAPE value of queue length. The fitness function takes the form:

Maximize: **Fitness function value (FV)** $= |\frac{1}{MAPE}|$ or,

Minimize: $\boldsymbol{MAPE} = \frac{1}{N} * \sum_{i=1}^{N} \boldsymbol{E}i$

iv. **Generation of the next population (Genetic Operation):** Based on various literatures, the maximum number of generations is chosen as 20. The process involves 3 operations using genetic operators. The genetic algorithm operation is shown in Figure 3-11 below.

- **Selection:** It is the process to determine the number of times an individual is selected from the current population for reproduction. The copies of better individuals will be reproduced from the fittest individuals to form the parent population for the next generation. It selects individuals with higher fitness values as they are more likely to contribute to the next generation. The different types of available selection scheme in the GA package include steady-state, roulette wheel, stochastic universal, rank, random, and tournament. Based on various literatures, the roulette wheel parent selection method is used.

- **Reproduction (Crossover):** Crossover is a basic genetic operator for producing new chromosomes. It is a recombination process that creates new individuals (offspring) in successive generations by combining material from a pair of selected individuals of the previous generation using crossover probability. The idea is to combine the characteristics of the parents to potentially produce better solutions. The different types of available crossover methods in the GA package include single point, two points, uniform, and scattered. Based on various literatures, the two-point crossover option is used and the probability of crossover is chosen as 0.8.

- **Mutation:** Mutation is another genetic operator to modify the elements of the chromosomes. It involves the addition of new information in a random way to the offspring which ultimately helps to avoid getting trapped at local optima using mutation probability. Mutation helps prevent premature convergence to suboptimal solutions and enables the exploration of some of the areas that have not been searched. The different types of available mutation methods in the GA package include random, swap, inversion, scramble, and adaptive. Based on various literatures, the random mutation option is used and the probability of mutation is chosen as 0.05.
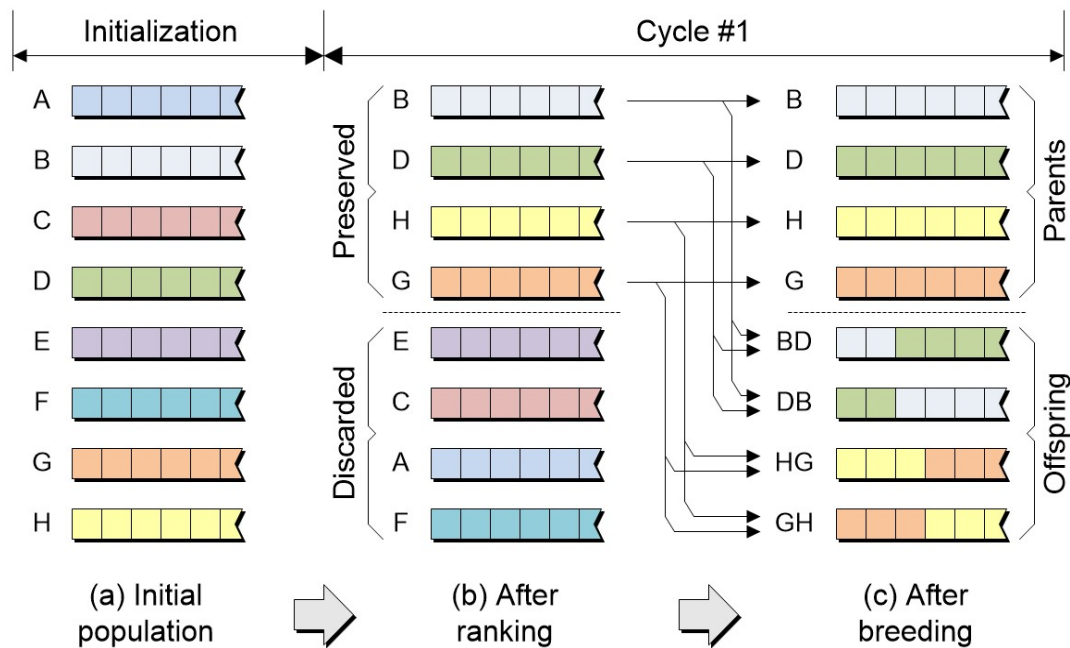
Figure 3-11 Genetic Algorithm Operation (Source: Max Maxfield)

v. **Replacement (Reinsertion):** This process involves the replacement of the parents of the previous generation by their offspring bringing a new set of the more fit population for the next generation. The replacement is based on the fitness function values of the individuals.

vi. **Termination (Stopping criteria):** The algorithm terminates when a stopping criterion is met, which could be a maximum number of generations, a target fitness value, saturation of the target fitness value, or a specific amount of computational time. The stopping criteria employed in this study is the maximum number of generations which is set as 20. After the termination of the algorithm, the queue length is checked based on the MAPE threshold value. If the error in queue length is within the MAPE threshold value, the program is terminated. Otherwise, the program is again run using queue length as a fitness function.

"The control parameters for GA such as crossover rate, mutation rate, population size, and number of generations are critical to solution quality. Especially sufficient but not excessive chromosomes should be initialized to fill the population" (Ma and Abdulhai 2002). The different run time configuration settings used during the implementation of GA in the code are adopted based on values used in relevant literature (Pattnaik, Mohan et al. 1998, Ma and Abdulhai 2002, Park and Qi 2005, Park, Won et al. 2006, Mathew and Radhakrishnan 2010, Manjunatha, Vortisch et al. 2013, Siddharth and Ramadurai 2013) and is shown in Table 3-9.

Table 3-9 Adopted GA Run-Time Configuration Settings

S.N.	GA Run-Time Settings	Value
1	Number of generations	20
2	Population Size	20
3	Number of mating parents	10
4	Probability of Crossover	0.8
5	Probability of Mutation	0.05
6	Gene Type Representation	Float
7	Crossover Option	Two Points
8	Mutation Option	Random
9	Parent Selection	Roulette Wheel

3.8 Validation

Validation is a process to confirm the reliability of the calibrated model which is done by providing a different input data set to the calibrated model and checking its accuracy. After finding the optimal value of the sensitive parameters from calibration, the simulation models are then validated using a new set of field data including input traffic volume, traffic composition, etc. The validation of the model is done using the peak hour traffic data from the third day data of the three-day traffic count secondary data at each intersection. Using the calibrated parameter values, the simulation models are run for 5 random seeds. Input traffic flows are compared with the simulation output using MAPE values to check the reliability of the calibrated model.

Along with traffic volume, queue length is selected as an additional measure of effectiveness in validation. The extracted traffic data from the video graphic survey is used to validate the traffic flow for two sub intervals (30 minutes) within the peak hour and the maximum queue length within the peak hour. Based on studied literatures, many researchers have used queue length as an effective MOE for calibration and validation process (Park, Won et al. 2006, Jayasooriya and Bandara 2018, Shrestha 2022). "Further, the main justification for using queue length is the practicality and accuracy of obtaining data as opposed to obtaining vehicle or passenger delays in the practical

context" (Jayasooriya and Bandara 2018). So, the models are also validated by comparing field values of maximum queue length with those obtained from VISSIM using MAPE values. This is done to improve the reliability of the simulation models to the actual field conditions. Field values of queue length are measured at all approaches of the intersections for comparison with the simulated output. The queue length field data is recorded for the whole peak hour identified at all approaches of the individual intersections. The end of the queue line is marked and numbered serially at the start of green interval for each signal phase during the peak hour. After the completion of the peak hour, the queue length is then measured between the intersection stop line and the marked end line using a 50m measuring tape and noted down.

The longest queue length at each approach leg in the field is designated as the maximum queue length. The simulated queue length is obtained from VISSIM by using queue counters placed at the stop line of the model for each approach. The simulated maximum queue lengths of vehicles are obtained from the simulation models during the peak hour for each approach. "A variety of techniques have been used by researchers to validate the calibrated models such as visual validation, statistical validation using two-sample tests, absolute percentage error, and mean absolute percentage error" (Dutta and Ahmed 2019). But similar to the whole research, the field maximum queue lengths are compared with the simulation output using MAPE for validation. "The simulation model is considered to be validated if the MAPE values are within acceptable limits of 15%" (Dowling, Skabardonis et al. 2004, Dutta and Ahmed 2019).

CHAPTER 4.   RESULTS AND DISCUSSION

4.1   VISSIM Model Formation

The first step towards the research analysis involves the formation of the VISSIM model of the study area intersections as explained in the Section 3.3.4. The VISSIM network was created using the primary source of field data and secondary data. The approach legs were drawn using links with appropriate lane dimensions and they were connected with each other using connectors for the different directional movements. The approach links were made long enough to the point that they reached the next intersection so that the vehicle flow and queue length generated would be more accurate. Figure 4-1 and Figure 4-2 shows the formulated VISSIM model of Singhadurbar intersection in 2D and 3D visualization respectively. Similarly, Figure 4-3, Figure 4-4, Figure 4-5 and Figure 4-6 shows the 2D and 3D visualization of Padmodaya and Putalisadak intersections respectively.

Figure 4-1 Screenshot of the Simulation Run of Singhadurbar Intersection in VISSIM (2D Visualization)
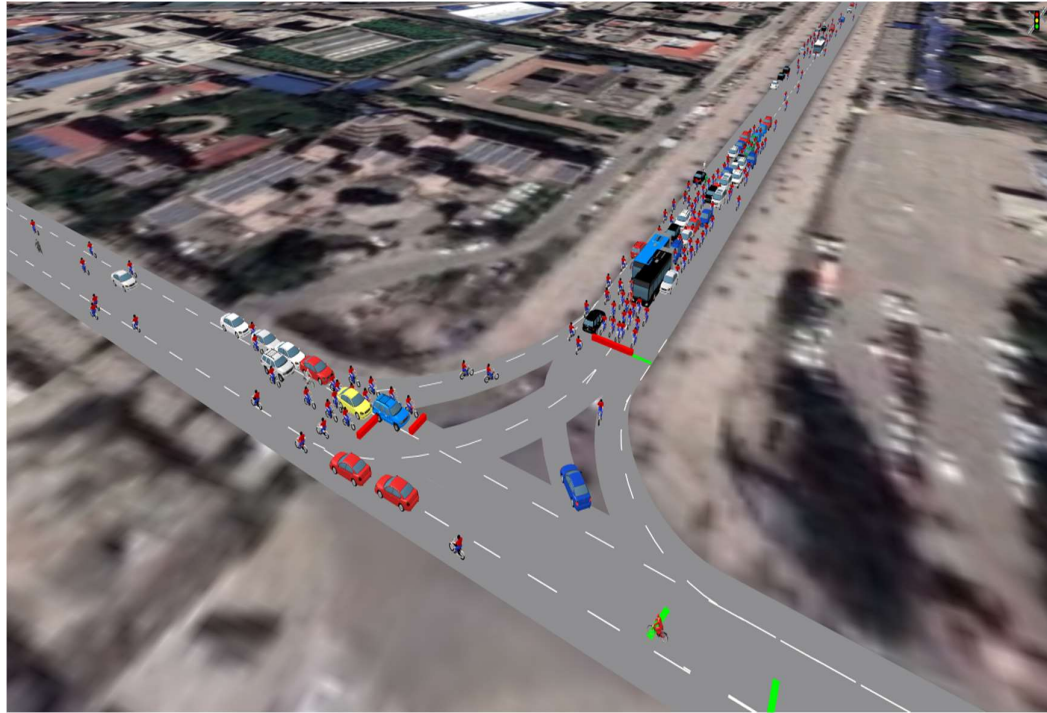
Figure 4-2 Screenshot of the Simulation Run of Singhadurbar Intersection in VISSIM (3D Visualization)

Figure 4-3 Screenshot of the Simulation Run of Padmodaya Intersection in VISSIM (2D Visualization)

Figure 4-4 Screenshot of the Simulation Run of Padmodaya Intersection in VISSIM (3D Visualization)

Figure 4-5 Screenshot of the Simulation Run of Putalisadak Intersection in VISSIM (2D Visualization)

Figure 4-6 Screenshot of the Simulation Run of Putalisadak Intersection in VISSIM (3D Visualization)

4.2 Data Analysis and Extraction

The data analysis based on various literature review are summarized below.

1. Vehicle Types - The data analysis and the adopted vehicle types are provided in the Section 3.3.2 and Table 3-3 above respectively.

2. Vehicle Dimensions - The data analysis and the adopted average vehicles dimensions are provided in the Section 3.3.4.2 and Table 3-4 above respectively.

3. Desired Speed Distributions - The data analysis and the desired minimum and maximum speed of different vehicle categories are provided in the Section 3.3.4.6 and Table 3-5 above respectively.

4. Passenger Car Unit (PCU) - The data analysis and the adopted PCU values are provided in the Section 3.3.2 and Table 3-3 above respectively.

5. Traffic Representation - The adopted driver behavior parameter options for representing the traffic network are provided in the Section 3.3.4.3.

6. Driver Behavior Parameters – The literature regarding the various driving behavior parameters studied are provided in the Section 3.4.

4.2.1 Peak Hour

The literature regarding the peak hour analysis are provided in the Section 3.3.3. The 24 hours traffic volume count for 3 days of the study area intersections were analyzed at 15 minute intervals for the adopted vehicle categories. Using the adopted PCU factor, the peak hour time and the peak hour volume in vehicles/hr and PCU/hr were obtained. A sample of the total volume counts for 15 minute intervals for all 3 days including PCU values and the adopted peak hour time have been shown in Table 4-1, Table 4-2, and Table 4-3 for Singhadurbar, Putalisadak and, Padmodaya intersection respectively.

The peak hour in Putalisadak and Padmodaya intersections were found to be different at different days but it was the same in the Singhadurbar intersection. So, based on visual examination and personal engineering judgement to get the best representative data set, the peak hour for Singhadurbar, Putalisadak, and Padmodaya intersections were adopted as 10:00 AM – 11:00 AM, 10:15 AM – 11:15 AM, and 10:00 AM – 11:00 AM respectively.

Table 4-1 Peak Hour Analysis of Singhadurbar Intersection

Start Time (AM)	End Time (AM)	1st Day Vehicles	PCU	Peak Hour Volume (PCU/hr)	2nd Day Vehicles	PCU	Peak Hour Volume (PCU/hr)	3rd Day Vehicles	PCU	Peak Hour Volume (PCU/hr)	Adopted Peak Hour Time (AM)
09:00	09:15	1216	683.3		1286	725.3		1147	650.3		
09:15	09:30	1350	750.2		1512	838.8		1533	864.5		
09:30	09:45	1499	796.4		1646	884.3		1725	927.3		
09:45	10:00	1632	882.5		1840	993.1		1786	976.8		
10:00	10:15	1951	1102.9		2163	1231.1		2144	1221		
10:15	10:30	1905	1074.5	4293.2	2134	1192.2	4813	2024	1152.9	4709.4	10:00 – 11:00
10:30	10:45	1931	1094.3		2205	1241.7		2069	1185.2		
10:45	11:00	1881	1021.5		2128	1148		2104	1150.3		
11:00	11:15	1875	1031.7		2195	1200.9		2119	1174.9		
11:15	11:30	1799	972.2		2095	1113.3		2075	1134.6		
11:30	11:45	1756	941.8		2018	1074.1		1820	990.9		
11:45	12:00	1888	976		2185	1117.4		2081	1089.9		

Table 4-2 Peak Hour Analysis of Putalisadak Intersection

Start Time (AM)	End Time (AM)	1st Day		Peak Hour Volume (PCU/hr)	2nd Day		Peak Hour Volume (PCU/hr)	3rd Day		Peak Hour Volume (PCU/hr)	Adopted Peak Hour Time (AM)
		Vehicles	PCU		Vehicles	PCU		Vehicles	PCU		
09:00	09:15	1231	627.4		1533	802		1552	798.8		
09:15	09:30	1397	669.3		1633	804		1741	842.7		
09:30	09:45	1528	742.2		1813	901.5		1862	904		
09:45	10:00	1683	785.7		1972	948.1		2104	1016.8		
10:00	10:15	1859	877.4		2089	1010.7		2297	1122.4	4188.2	
10:15	10:30	1863	859.7		2111	999.7		2208	1023.7		
10:30	10:45	1847	866	3538	2131	1026		2148	1025.3		10:15 – 11:15
10:45	11:00	1876	874.2		2061	979.9	4110.2	1769	821.3		
11:00	11:15	1943	938.1		2151	1068.4		1666	774.5		
11:15	11:30	1833	853.7		2172	1035.9		1681	773.9		
11:30	11:45	1795	809.3		2110	980.1		1712	804.1		
11:45	12:00	1768	794.3		2068	954		1750	800.4		

Table 4-3 Peak Hour Analysis of Padmodaya Intersection

Start Time (AM)	End Time (AM)	1st Day		Peak Hour Volume (PCU/hr)	2nd Day		Peak Hour Volume (PCU/hr)	3rd Day		Peak Hour Volume (PCU/hr)	Adopted Peak Hour Time (AM)
		Vehicles	PCU		Vehicles	PCU		Vehicles	PCU		
09:00	09:15	1134	525.9		1373	646.9		1093	523.6		
09:15	09:30	1300	624.5		1505	724.6		1233	570.5		
09:30	09:45	1499	707.2		1665	797.8		1380	678.1		
09:45	10:00	1639	757.7		1766	835.8		1655	835.8		
10:00	10:15	1806	832		1908	923.1	3410.2	1772	839.6		
10:15	10:30	1790	807.4	3178	1796	834.1		2027	946.1		10:00 – 11:00
10:30	10:45	1750	777.3		1771	817.2		2017	894.4	3524.2	
10:45	11:00	1698	761.3		1758	814.9		1866	826.8		
11:00	11:15	1666	760.7		1734	811.5		1866	856.9		
11:15	11:30	1662	750.1		1753	833		1774	803		
11:30	11:45	1612	727.1		1677	775.9		1766	806.7		
11:45	12:00	1599	708		1649	735.1		1804	816.8		

4.2.2 Traffic Volume, Vehicle Inputs, and Compositions

The literature regarding the traffic volume analysis and vehicle composition analysis are provided in the Section 3.3.4.4. Sample data of the 24 hours traffic volume count for 3 days of the study area intersections analyzed at 15 minute intervals for the adopted vehicle categories are provided in APPENDIX A. The summary of the 24 hour traffic count analysis for 3 days of the approach legs in the chosen study area intersections are provided in APPENDIX B. The data showed that the average daily traffic and average PCU for Singhadurbar, Putalisadak, and Padmodaya intersections are 91417.67 (49952.93 PCU), 93478 (45140.03 PCU), and 80736.67 vehicles (38290.067 PCU) respectively. The data also showed that the maximum traffic occurred in the Sahidgate – Maitighar approach leg in Singhadurbar intersection, Hattisar – Singhadurbar approach leg in Putalisadak intersection, and Putalisadak – Singhadurbar approach leg in Padmodaya intersection. This may have occurred due to the location of major business and commercial centers around Putalisadak, Maitighar and Sahidgate area.

The vehicle input to VISSIM was provided in terms of vehicle numbers per hour of the peak hour data in each approach link of the model. The peak hour traffic volume data for day 1 and day 2 were averaged and given as input to the VISSIM model. The two day average peak hour traffic volume input in 15 minute intervals for calibration have been shown in Table 4-4, Table 4-5, and Table 4-6 for Singhadurbar, Putalisadak and Padmodaya intersection respectively.

Table 4-4 Traffic Volume Input for Calibration at Singhadurbar Intersection

Start Time (AM)	End Time (AM)	Two Day Average Traffic Volume Input for Calibration (veh)					
		M-S	M-P	P-S	P-M	S-P	S-M
Warmup Period		166	328	72	336	72	764
10:00	10:15	219	374	98	424	82	862
10:15	10:30	187	386	102	419	83	845
10:30	10:45	186	406	97	413	97	870
10:45	11:00	163	422	93	389	104	833
M-S = Maitighar to Sahidgate, M-P = Maitighar to Putalisadak, P-S = Putalisadak to Sahidgate, P-M = Putalisadak to Maitighar, S-P = Sahidgate to Putalisadak, and S-M = Sahidgate to Maitighar							

Table 4-5 Traffic Volume Input for Calibration at Putalisadak Intersection

Start Time (AM)	End Time (AM)	Two Day Average Traffic Volume Input for Calibration (veh)											
		B-D	B-H	B-S	D-B	D-H	D-S	S-D	S-B	S-H	H-D	H-B	H-S
Warmup Period		0	189	0	203	221	246	0	162	416	0	0	531
10:15	10:30	0	207	0	178	193	247	0	193	410	0	0	555
10:30	10:45	0	188	0	176	177	240	0	186	421	0	0	595
10:45	11:00	0	186	0	166	159	234	0	194	431	0	0	595
11:00	11:15	0	200	0	170	141	236	0	216	429	0	0	645
Similar to the above table, B = Bagbazar, D = Dillibazar, S = Singhadurbar, and H = Hattisar													

Table 4-6 Traffic Volume Input for Calibration at Padmodaya Intersection

Start Time (AM)	End Time (AM)	Two Day Average Traffic Volume Input for Calibration (veh)					
		S-A	S-P	P-A	P-S	A-S	A-P
Warmup Period		195	346	213	244	333	373
10:00	10:15	208	354	248	291	365	393
10:15	10:30	214	347	247	292	330	364
10:30	10:45	213	340	242	309	306	352
10:45	11:00	242	328	230	328	259	342
Similar to the above tables, A = Anamnagar, S = Singhadurbar, and P = Putalisadak							

The vehicle compositions for each approach leg of the intersections were found to be different from the overall vehicle composition of the entire network. So, vehicle proportions of different vehicle types of each approach leg were computed from the data and assigned in the VISSIM model using static routing decisions. The VISSIM input traffic composition of the approach legs of the three day traffic volume data for calibration have been shown in Table 4-7, Table 4-8, and Table 4-9 for Singhadurbar, Putalisadak, and Padmodaya intersection respectively.

Table 4-7 Traffic Composition of Approach Legs of Singhadurbar Intersection

Traffic Composition						
Vehicle Type	**M-S**	**M-P**	**P-S**	**P-M**	**S-P**	**S-M**
Heavy Truck	0.00%	0.00%	0.00%	0.00%	0.00%	0.05%
Light Truck	0.00%	0.17%	0.01%	0.16%	0.07%	0.27%
Big Bus	0.00%	0.00%	0.00%	0.00%	0.00%	0.16%
Mini Bus	5.37%	2.57%	0.04%	0.23%	0.02%	2.64%
Micro Bus	2.93%	0.58%	0.00%	0.44%	0.00%	2.29%
Car	**18.82%**	**20.08%**	**21.56%**	**18.39%**	**16.71%**	**19.48%**
Motorcycle	**65.05%**	**68.99%**	**71.23%**	**72.42%**	**73.79%**	**69.66%**
Utility Vehicles	1.77%	1.96%	0.00%	1.24%	1.80%	1.57%
4 Wheel Drive	4.37%	3.17%	4.83%	4.84%	4.38%	3.35%
3 Wheeler (Tempo)	0.00%	0.10%	0.00%	0.00%	2.86%	0.00%
Bicycle	1.70%	2.37%	2.33%	2.29%	0.37%	0.55%
Similar to the above tables, M = Maitighar, S = Sahidgate, and P = Putalisadak						

Table 4-8 Traffic Composition of Approach Legs of Putalisadak Intersection

Traffic Composition							
Vehicle Type	**B-H**	**D-B**	**D-H**	**D-S**	**S-B**	**S-H**	**H-S**
Heavy Truck	0.00%	0.06%	0.04%	0.06%	0.00%	0.03%	0.01%
Light Truck	0.01%	0.11%	0.07%	0.10%	0.06%	0.04%	0.00%
Big Bus	0.00%	0.00%	0.00%	0.00%	2.11%	1.74%	0.02%
Mini Bus	0.00%	6.34%	0.00%	0.04%	1.26%	0.26%	0.68%
Micro Bus	0.03%	0.83%	1.82%	0.01%	0.04%	0.03%	0.02%
Car	**10.45%**	**14.85%**	**18.76%**	**19.14%**	**14.82%**	**16.95%**	**15.87%**
Motorcycle	**82.45%**	**70.69%**	**73.39%**	**71.25%**	**75.99%**	**71.19%**	**81.00%**
Utility Vehicles	0.61%	1.14%	0.30%	0.89%	0.28%	0.15%	0.23%
4 Wheel Drive	3.95%	2.66%	2.46%	2.20%	2.96%	7.34%	1.41%
3 Wheeler (Tempo)	0.05%	0.08%	0.13%	3.80%	0.00%	0.00%	0.09%
Bicycle	2.44%	3.24%	3.03%	2.49%	2.47%	2.27%	0.67%
Similar to the above tables, B = Bagbazar, D = Dillibazar, S = Singhadurbar, and H = Hattisar. The vehicle routes B-D, B-S, S-D, H-D and H-B have negligible traffic volume so their vehicle compositions were not required.							

Table 4-9 Traffic Composition of Approach Legs of Padmodaya Intersection

Traffic Composition						
Vehicle Type	**S-A**	**S-P**	**P-A**	**P-S**	**A-S**	**A-P**
Heavy Truck	0.00%	0.02%	0.00%	0.04%	0.08%	0.01%
Light Truck	0.01%	0.01%	0.02%	0.12%	0.10%	0.08%
Big Bus	0.00%	0.00%	0.00%	0.00%	0.00%	0.00%
Mini Bus	0.05%	2.14%	1.90%	0.27%	0.03%	1.71%
Micro Bus	0.00%	0.01%	0.08%	0.02%	0.01%	0.35%
Car	**20.17%**	**15.32%**	**12.21%**	**14.06%**	**17.62%**	**12.47%**
Motorcycle	**72.51%**	**78.15%**	**80.10%**	**75.15%**	**71.44%**	**77.18%**
Utility Vehicles	0.01%	0.36%	0.19%	0.08%	0.15%	0.54%
4 Wheel Drive	3.51%	2.50%	3.60%	7.56%	8.69%	3.23%
3 Wheeler (Tempo)	2.39%	0.00%	0.02%	0.00%	0.01%	1.88%
Bicycle	1.34%	1.48%	1.88%	2.69%	1.86%	2.53%
Similar to the above tables, A = Anamnagar, S = Singhadurbar, and P = Putalisadak						

The following Figure 4-7, Figure 4-8, and Figure 4-9 shows the vehicle composition of the entire intersection network of Singhadurbar, Putalisadak and Padmodaya respectively. The data shows that motorcycle contributes the most to the total traffic volume followed by car in all of the study area intersections. The vehicle composition follows almost the same pattern in all of the approach legs. The significant percentage difference between the privately owned vehicles and public ones is a concerning issue to the traffic network of Kathmandu Valley intersections.
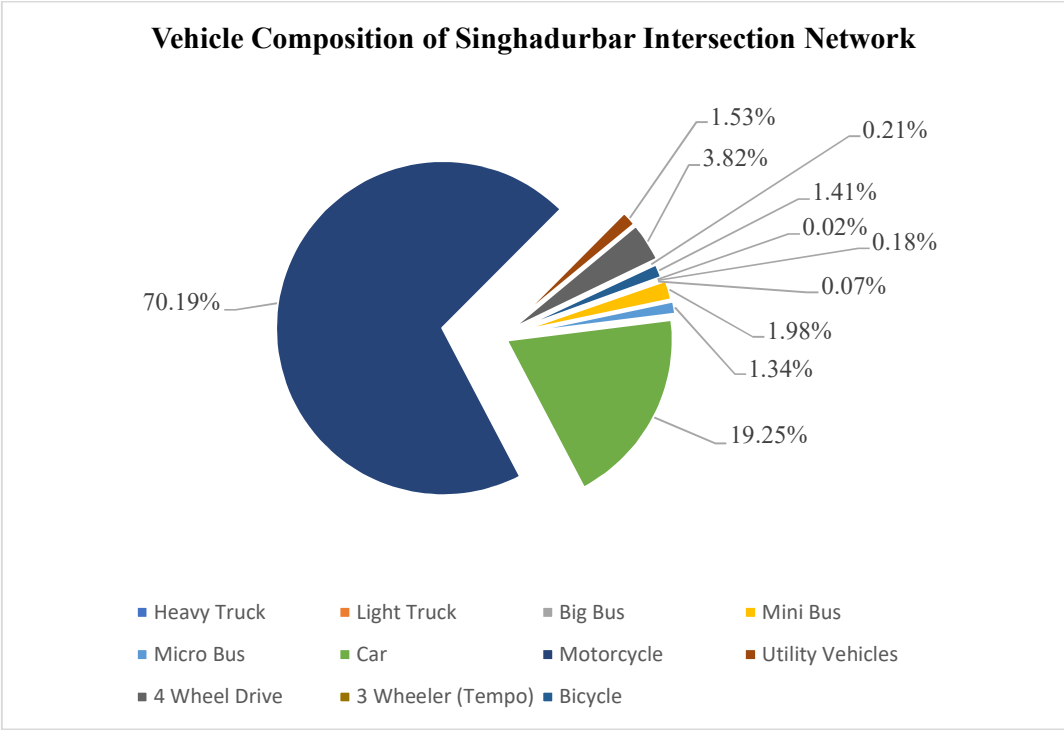
Figure 4-7 Vehicle Composition of the Singhadurbar Intersection Network
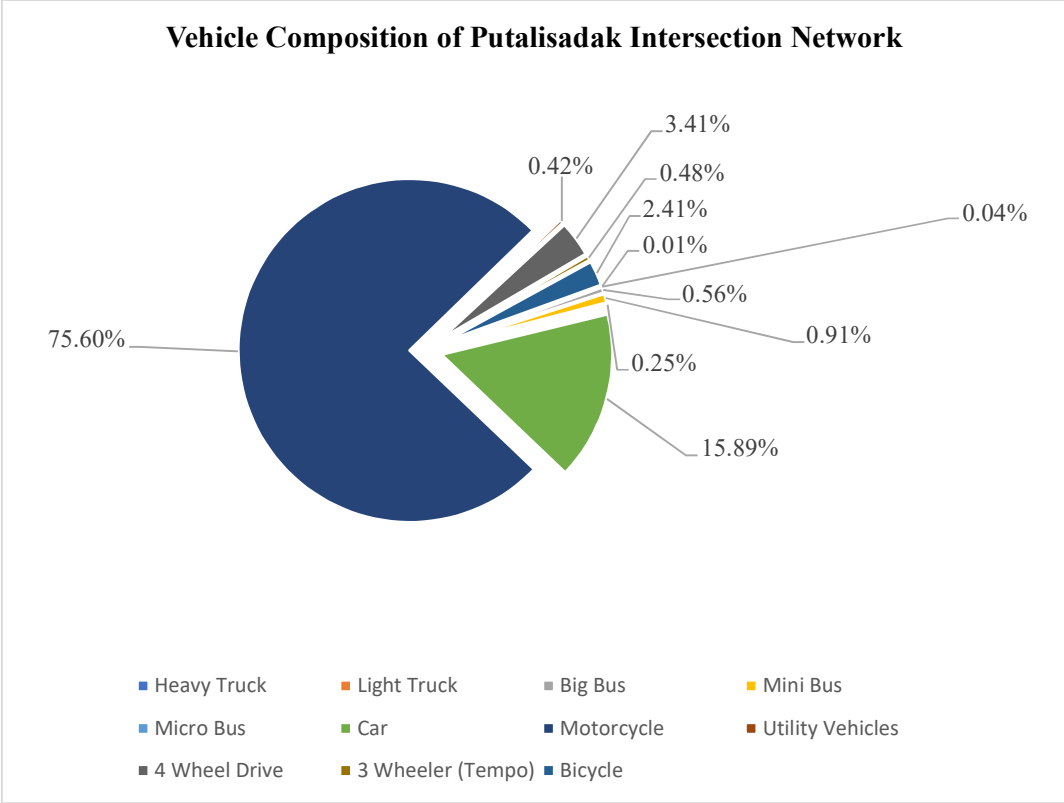
Figure 4-8 Vehicle Composition of the Putalisadak Intersection Network
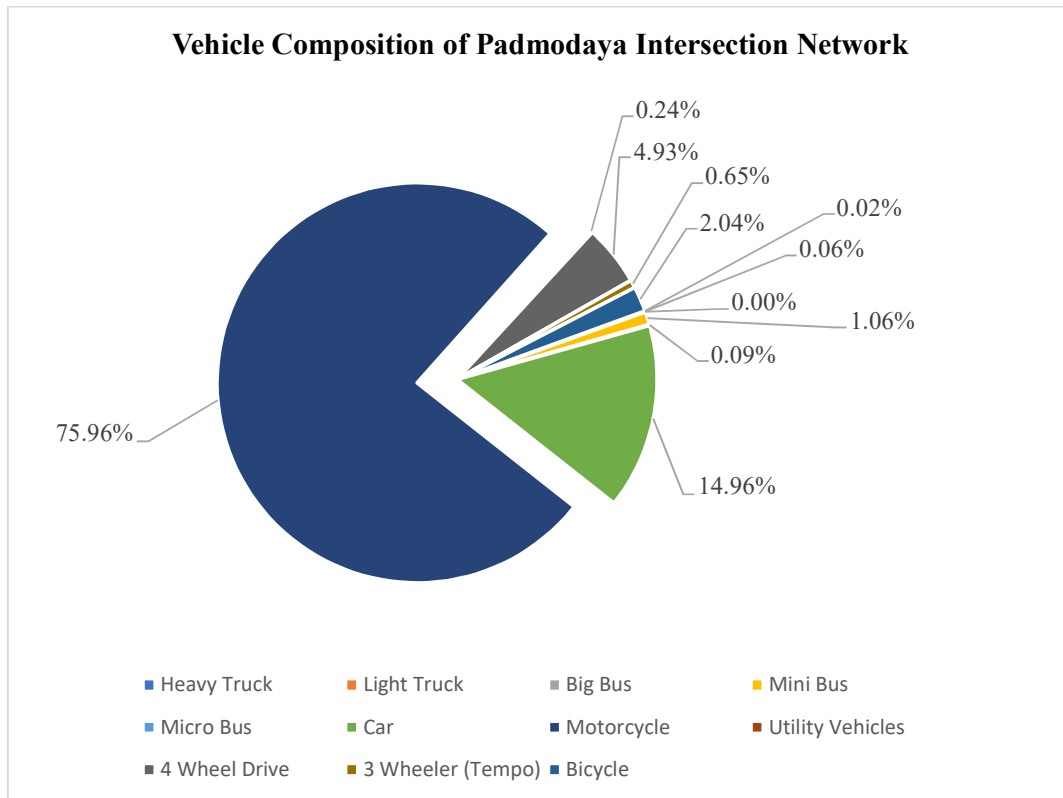
Figure 4-9 Vehicle Composition of the Padmodaya Intersection Network

4.2.3 Vehicle Routing and Relative Flows

The literature regarding the vehicle routing analysis are provided in the Section 3.3.4.4. The summary of the vehicle routing and relative flow analysis for 3 days of the approach legs in the chosen study area intersections are provided in APPENDIX C. The VISSIM input vehicle routing and relative flows of the approach legs based on two day average peak hour traffic volume in 15 minute intervals for calibration have been shown in Table 4-10, Table 4-11, and Table 4-12 for Singhadurbar, Putalisadak, and Padmodaya intersection respectively. The relative flow for B-D, B-S, S-D, H-D and H-B vehicle routes in Putalisadak intersection are taken as zero due to their negligible traffic volume.

Table 4-10 Vehicle Routing Decision and Relative Flow of Singhadurbar Intersection

Intersection Approach Legs	Vehicle Routing Decision	Relative Flow – As per defined time interval in seconds				
		Warmup Period (0 - 300)	300 - 1200	1200 - 2100	2100 - 3000	3000 - 3900
Maitighar	M-S	0.336	0.369	0.327	0.315	0.28
	M-P	0.665	0.632	0.674	0.686	0.721
Putalisadak	P-S	0.177	0.188	0.196	0.19	0.193
	P-M	0.824	0.813	0.805	0.811	0.808
Sahidgate	S-P	0.086	0.087	0.09	0.101	0.111
	S-M	0.915	0.914	0.911	0.9	0.89

Table 4-11 Vehicle Routing Decision and Relative Flow of Putalisadak Intersection

Intersection Approach Legs	Vehicle Routing Decision	Relative Flow – As per defined time interval in seconds				
		Warmup Period (0 - 300)	300 - 1200	1200 - 2100	2100 - 3000	3000 - 3900
Bagbazar	B-D	0	0	0	0	0
	B-H	1	1	1	1	1
	B-S	0	0	0	0	0
Dillibazar	D-B	0.303	0.288	0.297	0.297	0.311
	D-H	0.331	0.313	0.299	0.285	0.258
	D-S	0.368	0.4	0.406	0.419	0.432
Singhadurbar	S-D	0	0	0	0	0
	S-B	0.28	0.321	0.306	0.311	0.335
	S-H	0.721	0.68	0.695	0.69	0.666
Hattisar	H-D	0	0	0	0	0
	H-B	0	0	0	0	0
	H-S	1	1	1	1	1

Table 4-12 Vehicle Routing Decision and Relative Flow of Padmodaya Intersection

Intersection Approach Legs	Vehicle Routing Decision	Relative Flow – As per defined time interval in seconds				
		Warmup Period (0 - 300)	300 - 1200	1200 - 2100	2100 - 3000	3000 - 3900
Singhadurbar	S-A	0.36	0.37	0.382	0.385	0.425
	S-P	0.641	0.631	0.619	0.616	0.576
Putalisadak	P-A	0.466	0.461	0.459	0.44	0.413
	P-S	0.535	0.54	0.542	0.561	0.588
Anamnagar	A-S	0.473	0.482	0.476	0.465	0.431
	A-P	0.528	0.519	0.525	0.536	0.57

4.2.4 Signal Control

The literature regarding the signal control analysis are provided in the Section 3.3.4.5. The signal groups were taken as those intervals when vehicles on a particular direction were stopped and were input to the VISSIM model. It was observed that there were 3 phases of movement in the Singhadurbar and Padmodaya intersections while the Putalisadak intersection consisted of 2 phases of movement. Green time was found to be followed by three seconds of amber time. The VISSIM input signal control for the different signal phases at 15 minute intervals have been shown in Table 4-13, Table 4-14, and Table 4-15 for Singhadurbar, Putalisadak, and Padmodaya intersection respectively.

Table 4-13 Existing Signal Phases and Timing at Singhadurbar Intersection

Time Interval (Sec)	Phase	Vehicle Route	Average Green Time (Sec)	Amber Time (Sec)	Cycle Time (Sec)
Warmup Period (0 – 300)	1	M-P, M-S, P-M, and S-P	The signal timing was kept the same for warmup period as 300 – 1200 time interval.		
	2	M-S, P-M, P-S, and S-P			
	3	M-S, S-P, and S-M			
300 – 1200	1	M-P, M-S, P-M, and S-P	112	3	321 (3 cycles)
	2	M-S, P-M, P-S, and S-P	59	3	
	3	M-S, S-P, and S-M	141	3	
1200 – 2100	1	M-P, M-S, P-M, and S-P	94	3	220 (4 cycles)
	2	M-S, P-M, P-S, and S-P	27	3	
	3	M-S, S-P, and S-M	90	3	
2100 – 3000	1	M-P, M-S, P-M, and S-P	117	3	290 (3 cycles)
	2	M-S, P-M, P-S, and S-P	27	3	
	3	M-S, S-P, and S-M	137	3	
3000 - 3900	1	M-P, M-S, P-M, and S-P	110	3	280 (3 cycles)
	2	M-S, P-M, P-S, and S-P	38	3	
	3	M-S, S-P, and S-M	123	3	

Table 4-14 Existing Signal Phases and Timing at Putalisadak Intersection

Time Interval (Sec)	Phase	Vehicle Route	Average Green Time (Sec)	Amber Time (Sec)	Cycle Time (Sec)
Warmup Period (0 – 300)	1	B-H, D-S, H-S, S-B, and S-H	The signal timing was kept the same for warmup period as 300 – 1200 time interval.		
	2	B-H, D-S, D-B, D-H, and S-B			
300 – 1200	1	B-H, D-S, H-S, S-B, and S-H	102	3	192 (4 cycles)
	2	B-H, D-S, D-B, D-H, and S-B	84	3	
1200 – 2100	1	B-H, D-S, H-S, S-B, and S-H	138	3	262 (4 cycles)
	2	B-H, D-S, D-B, D-H, and S-B	118	3	

2100 – 3000	1	B-H, D-S, H-S, S-B, and S-H	111	3	218 (4 cycles)
	2	B-H, D-S, D-B, D-H, and S-B	101	3	
3000 - 3900	1	B-H, D-S, H-S, S-B, and S-H	118	3	244 (4 cycles)
	2	B-H, D-S, D-B, D-H, and S-B	120	3	

Table 4-15 Existing Signal Phases and Timing at Padmodaya Intersection

Time Interval (Sec)	Phase	Vehicle Route	Average Green Time (Sec)	Amber Time (Sec)	Cycle Time (Sec)
Warmup Period (0 – 300)	1	S-A, S-P, P-A, and A-S	The signal timing was kept the same for warmup period as 300 – 1200 time interval.		
	2	S-P, P-A, P-S, and A-S			
	3	S-P, P-A, A-P, and A-S			
300 – 1200	1	S-A, S-P, P-A, and A-S	103	3	345 (3 cycles)
	2	S-P, P-A, P-S, and A-S	112	3	
	3	S-P, P-A, A-P, and A-S	121	3	
1200 – 2100	1	S-A, S-P, P-A, and A-S	102	3	303 (3 cycles)
	2	S-P, P-A, P-S, and A-S	85	3	
	3	S-P, P-A, A-P, and A-S	107	3	
2100 – 3000	1	S-A, S-P, P-A, and A-S	84	3	250 (3 cycles)
	2	S-P, P-A, P-S, and A-S	73	3	
	3	S-P, P-A, A-P, and A-S	84	3	
3000 - 3900	1	S-A, S-P, P-A, and A-S	113	3	318 (3 cycles)
	2	S-P, P-A, P-S, and A-S	108	3	
	3	S-P, P-A, A-P, and A-S	88	3	

The Red – Green – Amber signal state sequence has been used in the VISSIM input except for some approaches where permanent green sequence was applied for the continuous movement. The following Figure 4-10, Figure 4-11, and Figure 4-12 shows the signal phase movement diagrams of the entire intersection network of Singhadurbar, Putalisadak, and Padmodaya respectively. The approaches which are open throughout the signal cycle have been identified in the figures below using the double arrow.
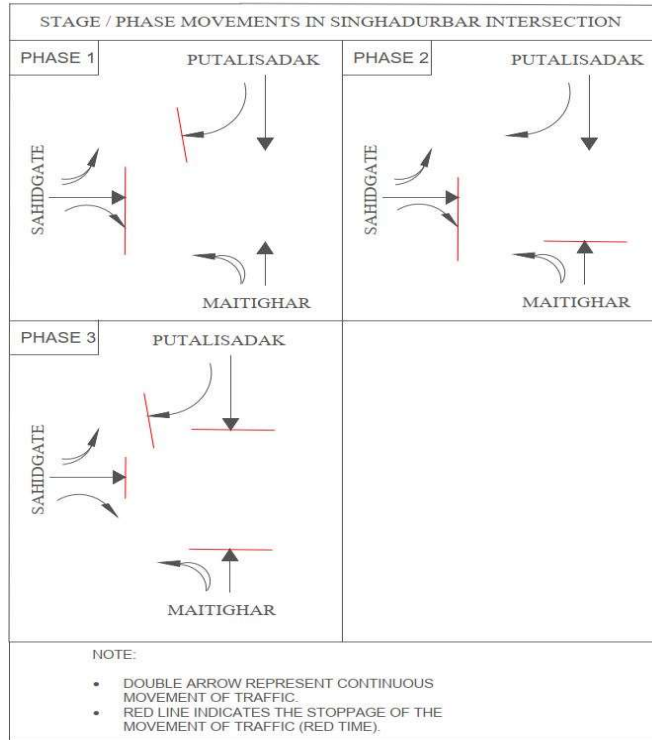
Figure 4-10 Signal Phase Movement Diagram of Singhadurbar Intersection
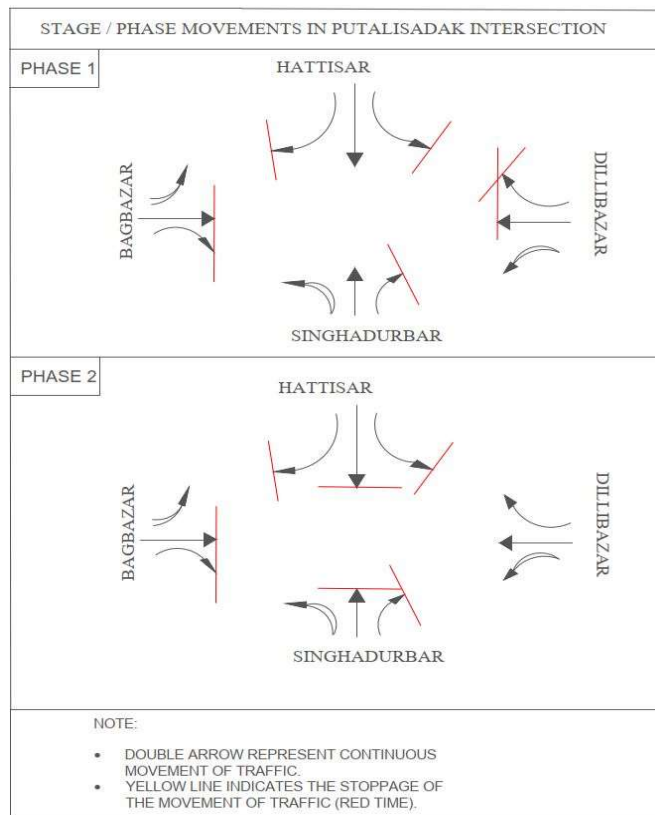
Figure 4-11 Signal Phase Movement Diagram of Putalisadak Intersection
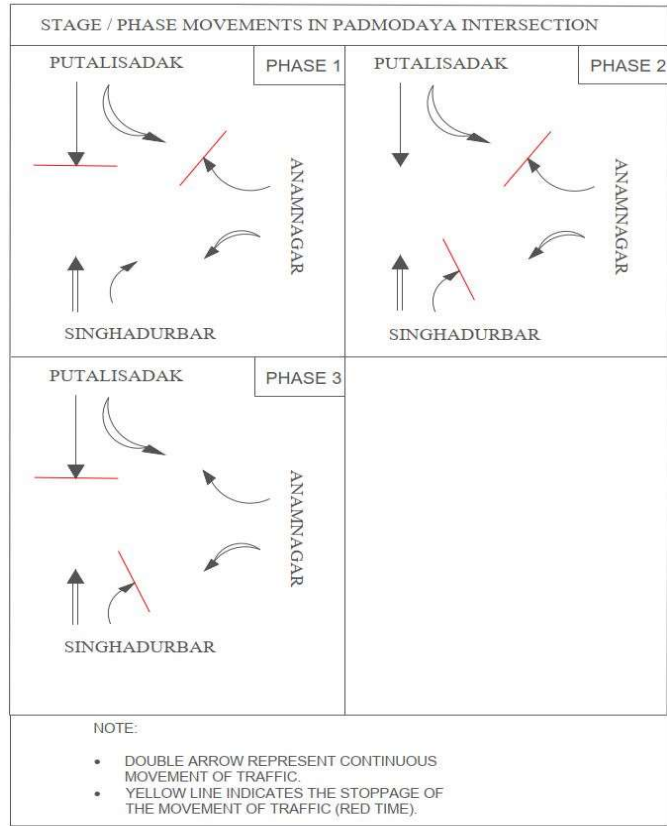
Figure 4-12 Signal Phase Movement Diagram of Padmodaya Intersection

4.3 Sensitivity Analysis

The literature regarding the sensitivity analysis are provided in the Section 3.5 and 3.6 and its subsequent subheadings 3.6.1, 3.6.2, 3.6.3, and 3.6.4. The sensitivity analysis was performed on all three intersections.

4.3.1 Initial Run on Default Parameters

The VISSIM model of the intersection under study was run on default parameter settings with 5 different random seed value. The VISSIM output and the percentage error in MAPE have been shown in Table 4-16, Table 4-17, and Table 4-18 for Sahidgate, Maitighar, and Putalisadak approach legs of the Singhadurbar intersection respectively. Similarly, the summary of the percentage error in MAPE for the three intersections have been shown in Table 4-19 below.

73

Table 4-16 Error for Default Parameter Values for Sahidgate Approach Leg of Singhadurbar Intersection

1. Sahidgate Approach Leg			
Time Period (s)	**Actual Flow (vehicles)**	**Flow with Default Parameters (vehicles)**	**Percentage Error (MAPE)**
0-300	Warmup Period		
300-1200	219	176	19.63%
	98	81	17.35%
1200-2100	187	148	20.86%
	102	85	16.67%
2100-3000	186	191	2.69%
	97	107	10.31%
3000-3900	163	126	22.70%
	93	87	6.45%
		Average MAPE	**14.58%**

Table 4-17 Error for Default Parameter Values for Maitighar Approach Leg of Singhadurbar Intersection

2. Maitighar Approach Leg			
Time Period (s)	**Actual Flow (vehicles)**	**Flow with Default Parameters (vehicles)**	**Percentage Error (MAPE)**
0-300	Warmup Period		
300-1200	424	391	7.78%
	862	500	42.00%
1200-2100	419	384	8.35%
	845	654	22.60%
2100-3000	413	453	9.69%
	870	493	43.33%
3000-3900	389	374	3.86%
	833	625	24.97%
		Average MAPE	**20.32%**

Table 4-18 Error for Default Parameter Values for Putalisadak Approach Leg of Singhadurbar Intersection

3. Putalisadak Approach Leg			
Time Period (s)	**Actual Flow (vehicles)**	**Flow with Default Parameters (vehicles)**	**Percentage Error (MAPE)**
0-300	Warmup Period		
300-1200	82	91	10.98%
	374	286	23.53%
1200-2100	83	91	9.64%
	386	301	22.02%
2100-3000	97	91	6.19%
	406	357	12.07%
3000-3900	104	272	161.54%
	422	329	22.04%
		Average MAPE	**33.50%**

Table 4-19 Summary of Initial MAPE for Default Parameter Values

Measure of Effectiveness	Intersection	Approach Leg	Vehicle Route	MAPE	Average MAPE	GEH
Traffic Volume	Singhadurbar	Sahidgate (S)	M-S	16.47%	14.58%	4.315
			P-S	12.69%		1.549
		Maitighar (M)	P-M	7.42%	20.31%	1.067
			S-M	33.20%		21.333
		Putalisadak (P)	S-P	47.08%	33.50%	8.387
			M-P	19.91%		8.328
					22.8%	
	Padmodaya	Singhadurbar (S)	A-S	30.43%	20.81%	11.984
			P-S	11.20%		4.068
		Putalisadak (P)	A-P	29.63%	15.91%	12.386
			S-P	2.20%		0
		Anamnagar (A)	S-A	4.39%	3.50%	0.271
			P-A	2.61%		0.096
					13.41%	

	Putalisadak	Bagbazar (B)	S-B	4.46%	4.53%	0.071
			D-B	4.60%		0
		Hattisar (H)	S-H	9.16%	7.84%	3.884
			B-H	4.60%		0
			D-H	9.76%		0.039
		Singhadurbar (S)	D-S	1.98%	2.97%	0.032
			H-S	3.97%		0.020
					5.11%	
Maximum Queue Length	Singhadurbar	Sahidgate	-	-	**84.46%**	**9.945**
		Maitighar	-	-	**84.63%**	**9.901**
		Putalisadak	-	-	**48.80%**	**5**
					72.63%	
	Padmodaya	Singhadurbar	-	-	**65.48%**	**6.806**
		Putalisadak	-	-	**95.10%**	**8.789**
		Anamnagar	-	-	**72.64%**	**9.003**
					77.74%	
	Putalisadak	Dillibazar	-	-	**34.68%**	**6.329**
		Hattisar	-	-	**39.22%**	**3.517**
		Singhadurbar	-	-	**55.64%**	**3.211**
					43.18%	

4.3.2 First Level ANOVA Testing

The VISSIM simulations were performed on a HP Elite Book 830 G8 Notebook PC with an 11th Gen Intel (R) Core i7-1165G7 @ 2.80 GHz processor and 16.00 GB of RAM. Using the identified driving behavior parameters along with their ranges as shown in Table 3-6, multiple simulations were performed using Latin Hypercube Sampling (LHS) and a written Python programming code. A total of 1000 runs for 12 calibration parameters was performed for 200 scenarios generated by Latin Hypercube Sampling (LHS) for 5 random seeded runs which took 37 hours and 43 minutes for Singhadurbar, 35 hours and 40 minutes for Padmodaya, and 35 hours and 59 minutes for Putalisadak intersection. The 5 random seeded runs were then averaged to represent the results of each of the 200 parameter sets. The discrete values of the parameter from the LHS samples and the change in error with respect to the default parameter set values

are input into SPSS to perform one way ANOVA testing. Some samples of the outputs generated by SPSS for First Level One Way ANOVA Testing are attached in APPENDIX D.

The p – value results of the First Level one way ANOVA have been shown in Table 4-20 for the Sahidgate, Maitighar, and Putalisadak approach legs of the Singhadurbar intersection. The results confirm that the following six driving behavior parameters are sensitive in the first level ANOVA testing for Singhadurbar intersection, whose p-values are less than 0.2 in any of the trials (Siddharth and Ramadurai 2013, Dutta and Ahmed 2019).

1. Minimum Look Back Distance
2. Average Standstill Distance
3. Additive Part of Safety Distance
4. Multiplicative Part of Safety Distance
5. Minimum Clearance (Front/Rear)
6. Minimum Lateral Distance (Standing) at 0 km/h

Similarly, the results of the first level ANOVA testing for the Padmodaya and Putalisadak intersection have been shown in Table 4-21 below. The SPSS analysis of the Bagbazar and Singhadurbar approach legs of Putalisadak intersection were not performed because the SPSS input i.e. the change in error with respect to the default parameter set values were significantly small. These values could lead to erratic output hence, these data were omitted during the study.

Table 4-20 First Level ANOVA results of Singhadurbar

ANOVA Probability (p – value) of Singhadurbar Intersection						
Driving Behavior Parameter	**Default Values**	**Sahidgate**	**Maitighar**	**Putalisadak**	**Remarks**	**Result**
Minimum Look Ahead Distance (m)	0	0.257	0.274	0.335	All trials > 0.2	NS
Maximum Look Ahead Distance (m)	250	0.382	0.896	0.404	All trials > 0.2	NS
Minimum Look Back Distance (m)	**0**	0.395	**0.035**	0.275	**Trial 2 < 0.2**	**S**
Maximum Look Back Distance (m)	150	0.979	0.94	0.933	All trials > 0.2	NS
Average Standstill Distance (m)	**2**	**6.32357 E-31**	**3.65783 E-15**	**1.56049 E-39**	**All trials < 0.2**	**S**
Additive Part of Safety Distance	**2**	**0.000434458**	**2.04711 E-11**	**0.000119478**	**All trials < 0.2**	**S**
Multiplicative Part of Safety Distance	**3**	**0.001**	**1.68113 E-08**	**0.006**	**All trials < 0.2**	**S**
Waiting Time Before Diffusion (s)	60	0.43	0.292	0.374	All trials > 0.2	NS
Minimum Clearance (Front/Rear) (m)	**0.5**	**0.098**	0.234	**0.038**	**Trial 1 and 3 < 0.2**	**S**
Safety Distance Reduction Factor	0.6	0.586	0.845	0.706	All trials > 0.2	NS
Minimum Lateral Distance (Standing) at 0 km/h (m)	**0.2**	**0.0000005**	**0.0001555**	**0.0001788**	**All trials < 0.2**	**S**
Minimum Lateral Distance (Driving) at 50 km/h (m)	1	0.31	0.417	0.287	All trials > 0.2	NS
S = Sensitive and NS = Not Sensitive						

Table 4-21 First Level ANOVA results of Padmodaya and Putalisadak

Driving Behavior Parameter	Padmodaya				Putalisadak	
	Singhadurbar	Anamnagar	Putalisadak	Result	Hattisar	Result
Minimum Look Ahead Distance (m)	0.604	0.413	0.761	NS	**0.04**	**S**
Maximum Look Ahead Distance (m)	0.791	0.704	0.786	NS	0.355	NS
Minimum Look Back Distance (m)	0.476	0.386	0.406	NS	0.835	NS
Maximum Look Back Distance (m)	0.357	**0.172**	0.358	S	0.592	NS
Average Standstill Distance (m)	**8.61E-41**	**3.85E-34**	**1.00E-37**	S	**1.29E-28**	S
Additive Part of Safety Distance	**1.14E-12**	**7.37E-10**	**2.71E-12**	S	**1.59E-07**	S
Multiplicative Part of Safety Distance	0.573	0.711	0.327	NS	**5.67E-07**	S
Waiting Time Before Diffusion (s)	0.656	0.728	0.495	NS	0.857	NS
Minimum Clearance (Front/Rear) (m)	0.51	0.259	0.519	NS	0.409	NS
Safety Distance Reduction Factor	0.834	0.785	0.562	NS	0.506	NS
Minimum Lateral Distance (Standing) at 0 km/h (m)	**0.006**	**0.155**	**0.025**	S	**0.001**	S
Minimum Lateral Distance (Driving) at 50 km/h (m)	0.524	0.741	0.65	NS	**0.06**	**S**
S = Sensitive and NS = Not Sensitive						

(The header row spans "ANOVA Probability (p – value)" across all columns)

4.3.3 Second Level ANOVA Testing

Multiple VISSIM simulations were again performed using the same Python programming code with some minor adjustments by removing the sensitive parameters identified from the first level of testing. A total of 1000 runs now for 6 calibration parameters which were not identified as sensitive in the first level ANOVA testing, was performed for 200 scenarios generated by Latin Hypercube Sampling (LHS) for 5 random seeded runs which took 45 hours and 39 minutes in the case of Singhadurbar

intersection. 8 calibration parameters were used in the case of Padmodaya intersection which took 28 hours and 13 minutes, and 6 parameters were used in Putalisadak intersection which took 27 hours and 52 minutes. The process was repeated again similarly to the first level ANOVA testing process to perform one way ANOVA using SPSS. Some samples of the outputs generated by SPSS for Second Level One Way ANOVA Testing are attached in APPENDIX D.

The p – value results of the Second Level one way ANOVA have been shown in Table 4-22 for the Sahidgate, Maitighar, and Putalisadak approach legs of the Singhadurbar intersection respectively. The results showed that three driving behavior parameters which were not considered to be sensitive in the First Order sensitivity analysis are now found to be sensitive in the Second Order sensitivity analysis since the p-values for the parameters are less than 0.2 in all of the trials (Siddharth and Ramadurai 2013, Dutta and Ahmed 2019).

Similarly, the results of the second level ANOVA testing for the Padmodaya and Putalisadak intersection have been shown in Table 4-23 below. Based on the results of the first and second order ANOVA tests, nine parameters were found to be sensitive in Singhadurbar and Putalisadak, while seven parameters were found to be sensitive in Padmodaya intersection. The amalgamation of the sensitive parameters results in the identification of nine sensitive driving behavior parameters for VISSIM models for heterogeneous traffic conditions which are listed below.

1. Minimum Look Ahead Distance
2. Minimum Look Back Distance
3. Maximum Look Back Distance
4. Average Standstill Distance
5. Additive Part of Safety Distance
6. Multiplicative Part of Safety Distance
7. Minimum Clearance (Front/Rear)
8. Minimum Lateral Distance (Standing) at 0 km/h
9. Minimum Lateral Distance (Driving) at 50 km/h

Table 4-22 Second Level ANOVA results of Singhadurbar

ANOVA Probability (p – value) of Singhadurbar Intersection					
Driving Behavior Parameter	Sahidg ate	Maitig har	Putalis adak	Remarks	Result
Minimum Look Ahead Distance (m)	**3.56E-17**	**2.58E-10**	**3.78E-22**	**All trials < 0.2**	**S**
Maximum Look Ahead Distance (m)	0.799	0.861	0.601	All trials > 0.2	NS
Maximum Look Back Distance (m)	**0.052**	**0.122**	**0.042**	**All trials < 0.2**	**S**
Waiting Time Before Diffusion (s)	0.613	0.785	0.641	All trials > 0.2	NS
Safety Distance Reduction Factor	0.879	0.55	0.555	All trials > 0.2	NS
Minimum Lateral Distance (Driving) at 50 km/h (m)	**1.13E-11**	**1.03E-20**	**1.08E-12**	**All trials < 0.2**	**S**
S = Sensitive and NS = Not Sensitive					

Table 4-23 Second Level ANOVA results of Padmodaya and Putalisadak

ANOVA Probability (p – value)						
Driving Behavior Parameter	Padmodaya				Putalisadak	
	Singhadu rbar	Anamnag ar	Putalisad ak	Result	Hattisar	Result
Minimum Look Ahead Distance (m)	0.267	**0.109**	0.429	S	-	-
Maximum Look Ahead Distance (m)	0.601	0.409	0.523	NS	0.898	NS
Minimum Look Back Distance (m)	0.332	0.222	0.315	NS	**0.026**	S
Maximum Look Back Distance (m)	-	-	-	-	**0.007**	S
Multiplicative Part of Safety Distance	**1.66E-127**	**1.11E-138**	**6.38E-115**	S	-	-
Waiting Time Before Diffusion (s)	0.528	0.544	0.426	NS	0.663	NS
Minimum Clearance (Front/Rear) (m)	0.482	0.514	0.567	NS	**0.171**	S
Safety Distance Reduction Factor	0.486	0.634	0.348	NS	0.213	NS
Minimum Lateral Distance (Driving) at 50 km/h (m)	**0.083**	**0.086**	**0.078**	S	-	-
S = Sensitive and NS = Not Sensitive						

The result of the sensitivity analysis was compared to the different literatures reviewed during this study. It can be seen that the sensitivity of the Wiedemann - 74 car following parameters and lateral behavior parameters are consistent throughout different studies. The results for the following behavior parameters are also same throughout different studies except for maximum look back distance which was identified as a sensitive parameter in this study. The results of the lane change behavior parameters are different with the various reviewed literatures. Based on the comparisons, we can see that the results of the analyzed parameters of heterogeneous traffic conditions within Kathmandu are quite similar to the results of Indian heterogeneous traffic conditions. The discrepancy in the results of the sensitivity analysis with the various reviewed literatures are highlighted in Table 4-24 below.

Table 4-24 Comparison of Sensitivity Analysis Results with Various Literatures

Driving Behavior Parameter	Jayasooriya and Bandara 2018 (Sri Lanka)	Mathew and Radhakrishnan 2010 (India)	Manjunatha, Vortisch et al. 2013 (India)	Dutta and Ahmed 2019 (India)	Siddharth and Ramadurai, 2013 (India)	Ge and Menendez, 2012 (Zurich, Switzerland)
Minimum Look Ahead Distance (m)	Not Tested	Sensitive	Sensitivity analysis performed on Wiedemann 99 parameters (Not Tested)	Sensitive	Sensitive	Not Tested
Maximum Look Ahead Distance (m)	Not Tested	Not Sensitive		Not Sensitive	Not Tested	Not Tested
Minimum Look Back Distance (m)	Not Tested	Not Tested		Sensitive	Sensitive	Not Tested
Maximum Look Back Distance (m)	Not Tested	Not Tested		**Not Sensitive**	Not Tested	Not Tested
Average Standstill Distance (m)	Sensitive	Sensitive		Sensitive	Sensitive	Sensitive
Additive Part of Safety Distance	Sensitive	Sensitive		Sensitive	Sensitive	Sensitive
Multiplicative Part of Safety Distance	Sensitive	Sensitive		Sensitive	Sensitive	Sensitive
Waiting Time Before Diffusion (s)	Not Sensitive	**Sensitive**		Not Tested	Not Tested	Not Tested
Minimum Clearance (Front/Rear) (m)	**Not Sensitive**	Not Tested		Sensitive	Sensitive	**Not Sensitive**
Safety Distance Reduction Factor	Not Sensitive	Not Tested		Not Tested	Not Tested	**Sensitive**
Minimum Lateral Distance (Standing) at 0 km/h (m)	Sensitive	Sensitive		Sensitive	Sensitive	Not Tested
Minimum Lateral Distance (Driving) at 50 km/h (m)	Sensitive	Sensitive		Sensitive	Not Tested	Not Tested

83

4.4 Calibration

The literature regarding the automated calibration of VISSIM parameters are provided in the Section 3.7 and its subsequent subheadings 3.7.1, 3.7.2, and 3.7.3. In this study, traffic volume and queue length have been selected as the key measure of effectiveness for calibration. The traffic flow data used in calibration and queue length data were obtained in different periods of time. The volume counts observed in the primary data collection are similar to the secondary data and the traffic conditions during those periods are somewhat similar in signal timing, vehicle routing, geometric characteristics and seasonal conditions. Hence, the queue lengths obtained in the field were used in the automated calibration process as a check for MAPE values. The queue length was also being simultaneously checked during the calibration process. If the error in queue length is within the MAPE threshold value after the completion of the optimization process, the program is terminated. Otherwise, the program is again run using queue length as a fitness function. The initial MAPE values for traffic flow and queue length for the intersections were well above the recommended value of 15% as proposed by (Dowling, Skabardonis et al. 2004) and the GEH values were also more than 5 (ODOT 2011, FDOT 2014) as highlighted in the Table 4-19 above. Hence, calibration process was deemed necessary to reduce the errors for the accurate representation of the intersections.

The nine identified sensitive calibration parameters were adjusted to represent the field conditions of the intersection using the peak hour traffic data from the first and second day of the three-day traffic count data at each intersection. Multiple VISSIM simulations along with automated calibration of the nine sensitive driving behavior parameters with their ranges as shown in Table 3-6 were performed simultaneously using Latin Hypercube Sampling (LHS) and Genetic Algorithm in a written Python programming code. The first generation of the population was done using LHS to randomly permute the samples within each interval. The code was run with the configuration settings as defined in the Table 3-9. Automated calibration was done based on the selected measure of effectiveness i.e. traffic flow. A maximum of 400 runs was performed for 20 population and 20 generations. The code was run for the whole 20 generations. Among the 20 parent population, 10 population with the best fitness

value were chosen for genetic operation which resulted in further 20 offspring population. The 20 offspring population and the 10 parent population were again compared to find the best 10 population for further genetic operation. The whole process took 19 hours and 20 minutes, 8 hours and 8 minutes, and 10 hours and 33 minutes for the Singhadurbar, Padmodaya, and Putalisadak intersections respectively. The error in queue length were checked for the intersections after the completion of the optimization process which was found to be within the MAPE threshold value of 15%. Hence, the program was terminated instead of running the program again using queue length as a fitness function. The optimally best fit value of the sensitive parameters obtained are as shown in Table 4-25.

Table 4-25 Calibrated Values of the Sensitive Parameters

S. N.	Sensitive Calibration Parameters	Default Value	Calibrated Values			Recommended Range of Values
			Singha durbar	Padmo daya	Putali sadak	
1	Minimum Look Ahead Distance (m)	0	20	10	15	[10 – 20]
2	Minimum Look Back Distance (m)	0	18.71	15	16.24	[15 – 18.71]
3	Maximum Look Back Distance (m)	150	135.6	107.94	150	[107.94 – 150]
4	Average Standstill Distance (m)	2	1.5	0.3	0.88	[0.3 – 1.5]
5	Additive Part of Safety Distance	2	0.5	0.1	0.1	[0.1 – 0.5]
6	Multiplicative Part of Safety Distance	3	1	0	0	[0 – 1]
7	Minimum Clearance (Front/Rear) (m)	0.5	0.49	0.73	0.7	[0.49 – 0.73]
8	Minimum Lateral Distance (Standing) at 0 km/h (m)	0.2	0.2	0.41	0.4	[0.2 - 0.41]
9	Minimum Lateral Distance (Driving) at 50 km/h (m)	1	0.88	0.9	0.6	[0.6 – 0.9]

After the completion of the calibration process, the errors were recalculated. The MAPE values between the input and simulated flows for each individual approach legs and the average MAPE value of 6.83%, 4.29% and 3.91% for Singhadurbar, Padmodaya, and Putalisadak intersections respectively are acceptable and conform to the limits (15%). The reduction of MAPE value of traffic flow in every successive generation for Singhadurbar, Padmodaya, and Putalisadak intersections are shown in Figure 4-13, Figure 4-14, and Figure 4-15 below.
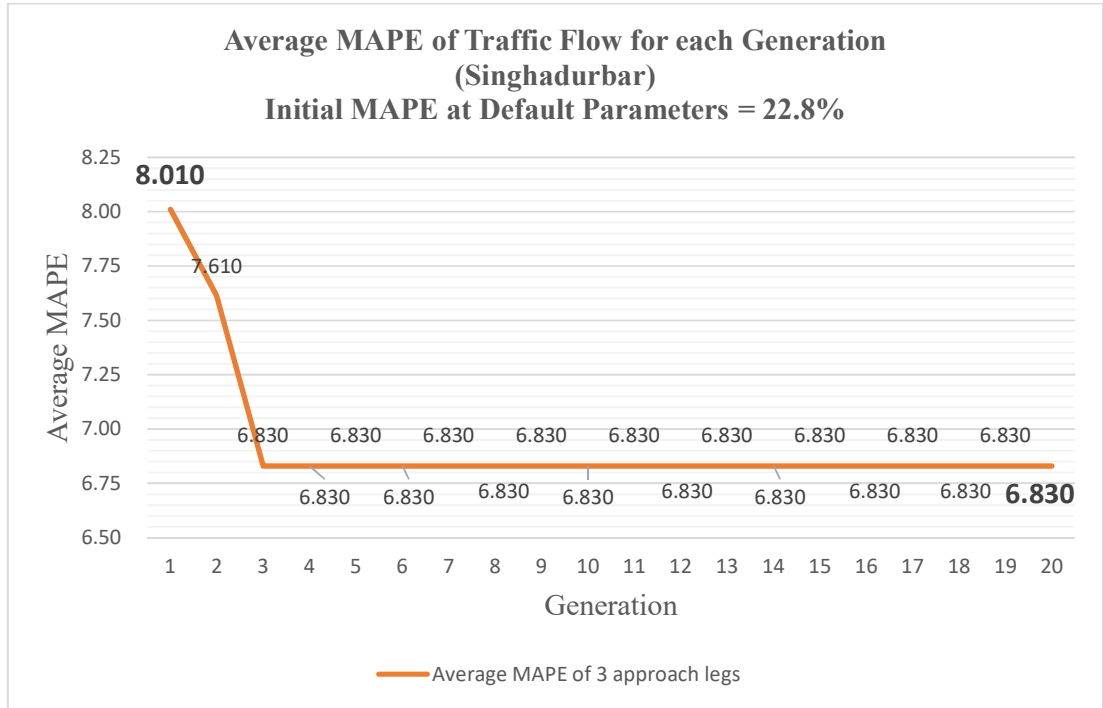
Figure 4-13 Average MAPE of Traffic Flow for each Generation of Singhadurbar Intersection
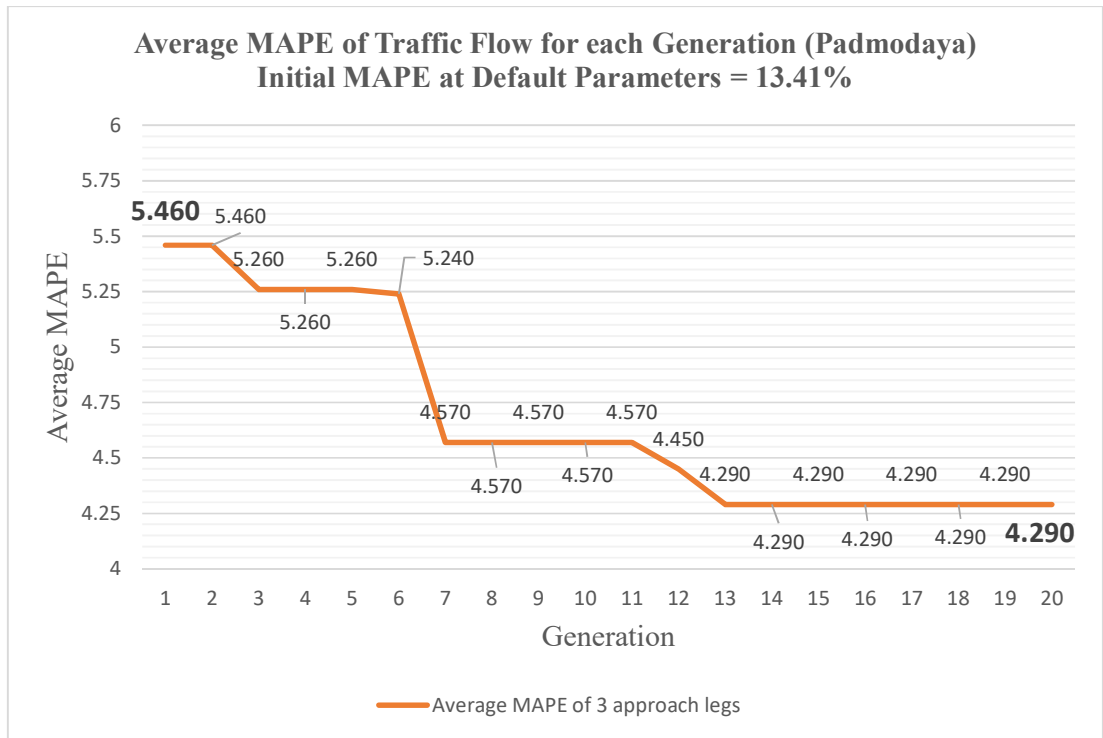
Figure 4-14 Average MAPE of Traffic Flow for each Generation of Padmodaya Intersection
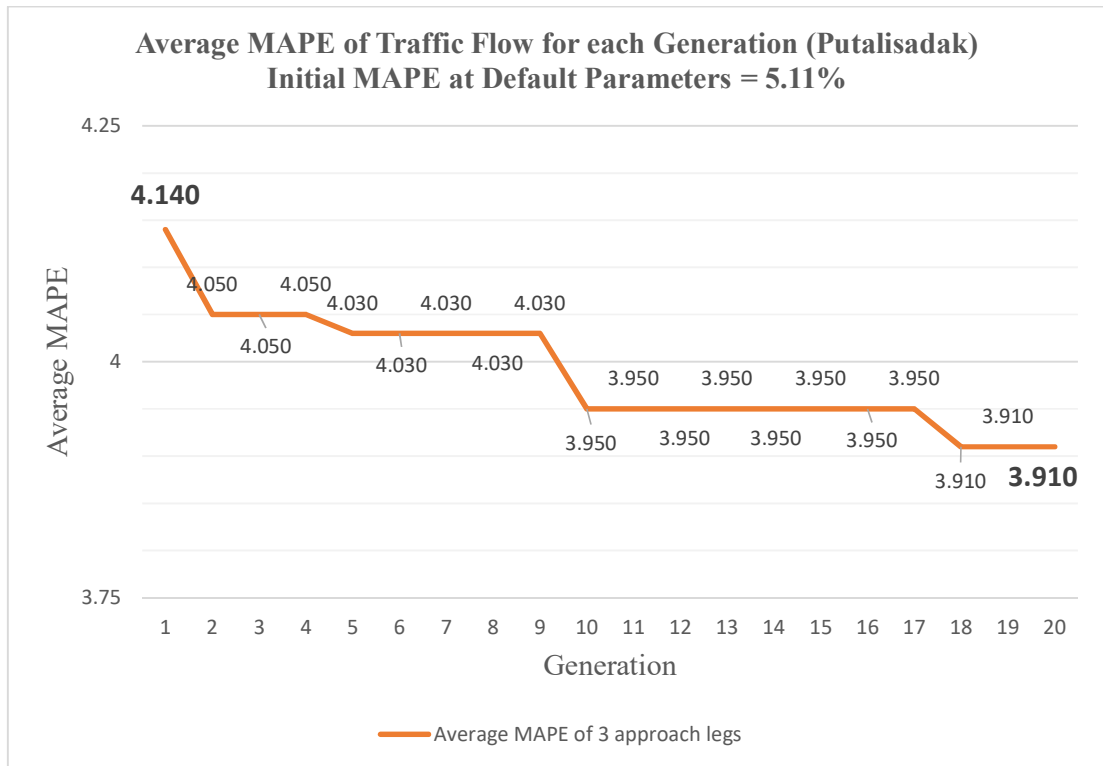
Figure 4-15 Average MAPE of Traffic Flow for each Generation of Putalisadak Intersection

Similarly, the errors in queue length were also calculated after calibration. The average MAPE between the field values and simulated values after calibration of 14.65%, 13.26% and 10.22% for Singhadurbar, Padmodaya, and Putalisadak intersections respectively are acceptable and conform to the limits (15%) as shown in Table 4-26 below.

Table 4-26 Average MAPE of Maximum Queue Length after Calibration

Interse ction	Approach Leg	Actual Maximum Queue Length	Maximum Queue Length in Calibrated Model	Average MAPE	GEH
Singha durbar	Sahidgate	230.53	270.360	17.28%	2.517
	Maitighar	198.13	150.436	24.07%	3.613
	Putalisadak	143.93	140.191	2.60%	0.317
				14.65%	
Padmo daya	Singhadurbar	143.417	107.042	25.36%	3.250
	Putalisadak	126.013	140.285	11.33%	1.237
	Anamnagar	209.41	215.866	3.08%	0.443
				13.26%	
Putalis adak	Dillibazar	120.66	109.030	9.64%	1.085
	Hattisar	80.177	75.586	5.73%	0.520
	Singhadurbar	165.365	190.680	15.31%	1.897
				10.22%	

The range of values for the sensitive calibration parameters obtained from the automated calibration process was compared to the different literatures reviewed during this study. It can be seen that the values of the Wiedemann - 74 car following parameters are within the recommended range throughout different studies. The values for the following behavior parameters are different with the various reviewed literatures but the values are very close to the end limits of the recommended range with a tolerance level of ±10 m. The values for the lateral behavior parameters and the lane change behavior parameters of the various reviewed literatures are somewhat within the recommended range. These values are also very close to the end limits of the recommended range with a tolerance level of ±0.1 m. Based on the comparisons, we can see that the calibration values adopted during studies in Kathmandu are within the recommended range of this study. We can also see that that the results of the analyzed parameters of heterogeneous traffic conditions within Kathmandu are quite similar to the results of Indian heterogeneous traffic conditions based on the calibration values. The discrepancy in the values of the calibration parameters of the various reviewed literatures are highlighted in Table 4-27 below.

Table 4-27 Comparison of Automated Calibration Results with Various Literatures

S. N.	Sensitive Calibration Parameters	Recommended Range of Values	Shrestha 2022 (Nepal)	Acharya and Marsani 2020 (Nepal)	Sharma 2016 (Nepal)	Jayasooriya and Bandara 2018 (Sri Lanka)	Mathew and Radhakrishnan 2010 (India)	Dutta and Ahmed 2019 (India)	Siddharth and Ramadurai, 2013 (India)
1	Minimum Look Ahead Distance (m)	[10 – 20]	20	30	27.91	-	127.79	28	27.91
2	Minimum Look Back Distance (m)	[15 – 18.71]	12	5	14.31	-	-	21	14.31
3	Maximum Look Back Distance (m)	[107.94 – 150]	100	-	100	-	-	-	-
4	Average Standstill Distance (m)	[0.3 – 1.5]	0.4	0.3	1	1.5	1.33	1.4	1
5	Additive Part of Safety Distance	[0.1 – 0.5]	0.1	0.19	0.2	1.5	0.28	0.24	0.2
6	Multiplicative Part of Safety Distance	[0 – 1]	0.8	0.71	0.78	2	0.16	0.19	0.78
7	Minimum Clearance (Front/Rear) (m)	[0.49 – 0.73]	0.4	0.5	0.5	-	-	0.25	0.11
8	Minimum Lateral Distance (Standing) at 0 km/h (m)	[0.2 - 0.41]	0.25, 0.3 and 0.4	-	-	0.5	0.3	0.22	0.62
9	Minimum Lateral Distance (Driving) at 50 km/h (m)	[0.6 – 0.9]	0.3, 0.5 and 0.75	-	-	0.75	0.42	0.5	-

4.5 Validation

The literature regarding the validation of the VISSIM models are provided in the Section 3.8. After calibration, the calibrated model was then validated using a new set of field data i.e. the third day data of the three-day traffic count data. The simulation models were run for 5 random seeds and the input traffic flows were compared with the simulation output using MAPE values and GEH statistic. The validation of the traffic flow are shown in Table 4-28, Table 4-29, and Table 4-30 below for the Singhadurbar, Padmodaya, and Putalisadak intersections respectively. The simulation models were considered to be validated as the MAPE values were within acceptable limits of 15% (Dowling, Skabardonis et al. 2004, Dutta and Ahmed 2019) and GEH values were less than 5 (ODOT 2011, FDOT 2014).

Table 4-28 Validation of Flow in Singhadurbar Intersection

Time Period (AM)	Approach Leg	Vehicle Route	MAPE	Average MAPE	GEH
10:00 - 11:00	Sahidgate (S)	M-S	3.47%	8.95%	0.211
		P-S	14.42%		0.646
	Maitighar (M)	P-M	2.61%	2.63%	0.640
		S-M	2.65%		1.489
	Putalisadak (P)	S-P	9.74%	7.52%	0.052
		M-P	5.31%		1.118
				6.37%	

Table 4-29 Validation of Flow in Padmodaya Intersection

Time Period (AM)	Approach Leg	Vehicle Route	MAPE	Average MAPE	GEH
10:00 - 11:00	Singhadurbar (S)	A-S	6.76%	6.90%	0.391
		P-S	7.04%		0.082
	Putalisadak (P)	A-P	7.39%	6.85%	0.301
		S-P	6.31%		0.026
	Anamnagar (A)	S-A	8.68%	7.03%	0.099
		P-A	5.38%		0
				6.93%	

Table 4-30 Validation of Flow in Putalisadak Intersection

Time Period (AM)	Approach Leg	Vehicle Route	MAPE	Average MAPE	GEH
10:15 - 11:15	Bagbazar (B)	S-B	0.92%	8.29%	0
		D-B	15.67%		0.535
	Hattisar (H)	S-H	4.35%	9.10%	0.071
		B-H	7.31%		0
		D-H	15.65%		0.463
	Singhadurbar (S)	D-S	17.71%	16.16%	0.063
		H-S	14.61%		0.021
				11.18%	

Similarly, the traffic flow data from the video graphic survey and maximum queue length were also validated with the simulation output using MAPE values and GEH statistic for each approach. The field queue length data were obtained at the peak hour identified from the analysis of the 3 day traffic count data at each intersection. The video graphic survey data validation of the Singhadurbar, Padmodaya, and Putalisadak intersections are shown in Table 4-31, Table 4-32, and Table 4-33 below respectively. The simulation models were considered to be validated as the average MAPE values were within acceptable limits of 15% (Dowling, Skabardonis et al. 2004, Dutta and Ahmed 2019) and GEH values were less than 5 (ODOT 2011, FDOT 2014). Some of the outputs on the approach legs were seen to be over the acceptable limit of 15% as highlighted in Table 4-31, Table 4-32, and Table 4-33 below. The possible reason for such outputs may have been the effect of the fluctuations of signal timing during the peak hour as traffic control was being done manually by the traffic police officers. As the signal timing is not fixed throughout the peak hour and depends majorly upon the operation of the traffic police officers, these errors may have been generated.

Table 4-31 Validation of Video Graphic Survey in Singhadurbar Intersection

Time Period (AM)	Approach Leg	Vehicle Route	MAPE	Average MAPE	GEH
Traffic Volume					
10:00 - 10:30	Sahidgate (S)	M-S	7.15%	14.18%	0.191
		P-S	21.21%		3.474
	Maitighar (M)	P-M	8.03%	10.62%	0.436
		S-M	13.20%		1.471
	Putalisadak (P)	S-P	0.00%	2.82%	0
		M-P	5.63%		0.579
				9.20%	
Maximum Queue Length					

Time Period (AM)	Approach Leg	Actual Maximum Queue Length	Maximum Queue Length in Calibrated Model	Average MAPE	GEH
10:00 - 10:30	Sahidgate	230.53	194.411	15.67%	2.478
	Maitighar	198.13	240.423	21.35%	2.856
	Putalisadak	143.93	147.637	2.58%	0.307
				13.20%	

Table 4-32 Validation of Video Graphic Survey in Padmodaya Intersection

Time Period (AM)	Approach Leg	Vehicle Route	MAPE	Average MAPE	GEH
Traffic Volume					
10:00 - 10:30	Singhadurbar (S)	A-S	2.87%	4.81%	0.579
		P-S	6.75%		0.045
	Putalisadak (P)	A-P	13.49%	9.71%	0.406
		S-P	5.94%		0.037
	Anamnagar (A)	S-A	10.47%	8.39%	0
		P-A	6.32%		0
				7.64%	

Maximum Queue Length					
Time Period (AM)	Approach Leg	Actual Maximum Queue Length	Maximum Queue Length in Calibrated Model	Average MAPE	GEH
10:00 - 10:30	Singhadurbar	143.417	113.510	20.85%	2.639
	Putalisadak	126.013	110.242	12.52%	1.451
	Anamnagar	209.41	208.511	0.43%	0.062
				11.27%	

Table 4-33 Validation of Video Graphic Survey in Putalisadak Intersection

Time Period (AM)	Approach Leg	Vehicle Route	MAPE	Average MAPE	GEH
Traffic Volume					
10:15 - 10:45	Bagbazar (B)	S-B	1.46%	2.01%	0
		D-B	2.55%		0.525
	Hattisar (H)	S-H	1.75%	10.65%	0.032
		B-H	15.53%		0
		D-H	14.68%		2.276
	Singhadurbar (S)	D-S	2.49%	3.56%	0
		H-S	4.62%		0.031
				5.41%	
Maximum Queue Length					
Time Period (AM)	Approach Leg	Actual Maximum Queue Length	Maximum Queue Length in Calibrated Model	Average MAPE	GEH
10:15 - 10:45	Dillibazar	120.66	146.760	21.63%	2.257
	Hattisar	80.177	74.031	7.67%	0.7
	Singhadurbar	165.365	179.961	8.83%	1.111
				12.71%	

CHAPTER 5. CONCLUSION AND RECOMMENDATION

5.1 Conclusion

This study was performed to identify the sensitive calibration parameters of VISSIM models and determine their range of values applicable for intersections for heterogeneous traffic conditions in Kathmandu using genetic algorithm. Three unsignalized intersections (Singhadurbar, Padmodaya, and Putalisadak) were chosen for the study. A study carried out by the Department of Roads to collect 72 hours traffic volume counts at intersections in Kathmandu was used to identify the peak hours which were observed to be from 10:00 AM – 11:00 AM, 10:00 AM – 11:00 AM, and 10:15 AM – 11:15 AM for the Singhadurbar, Padmodaya, and Putalisadak intersections respectively. A video graphic survey was also carried out at the studied intersections for two sub intervals (30 minutes) within the peak hour for the purpose of model validation.

The models of the intersection were built in the latest version of PTV VISSIM 2023 (SP 06) "Academic License". The collected primary and secondary data were used to represent the geometric, vehicular, heterogeneous traffic, routing, signal, and other conditions of the intersections. The models were initially run on default parameters of VISSIM which showed that the MAPE values for volume and queue length were greater than the recommended value of 15% and the GEH values were also more than 5. Hence, calibration process was deemed necessary for the accurate representation of the intersections.

Before calibration, sensitivity analysis was performed to identify the sensitive calibration parameters using Latin Hypercube Sampling (LHS) and one way ANOVA testing in two levels. The optimum value for these sensitive parameters were obtained by minimizing the error between the simulated and field measures using genetic algorithm. Automated calibration was done based on the selected measures of effectiveness i.e. traffic flow and queue length. Multiple VISSIM simulations along with automated calibration were performed simultaneously using LHS and Genetic

Algorithm in a written Python programming code. Finally, a range of values for the calibration parameters was recommended based on the results of the calibration process. The numerical results are summarized in the followings:

- Using traffic flow as a measure of effectiveness, the first level ANOVA testing resulted in 6 sensitive calibration parameters in Singhadurbar, 4 parameters in Padmodaya, and 6 parameters in Putalisadak intersection.

- Second level ANOVA testing performed for the 6 calibration parameters which were not identified as sensitive in the first level ANOVA testing resulted 3 additional sensitive parameters in Singhadurbar similar to Putalisadak intersection, while 4 additional sensitive parameters among 8 calibration parameters were identified in Padmodaya intersection. Based on the amalgamation of the results of the first and second order ANOVA tests, nine sensitive driving behavior parameters were identified as minimum look ahead distance, minimum look back distance, maximum look back distance, average standstill distance, additive part of safety distance, multiplicative part of safety distance, minimum clearance (front/rear), minimum lateral distance (standing) at 0 km/h, and minimum lateral distance (driving) at 50 km/h. So, the sensitive analysis identified a total of 9 calibration parameters sensitive in context of Kathmandu. Based on the comparisons, we can see that the results of the analyzed parameters of heterogeneous traffic conditions within Kathmandu are quite similar to the results of Indian heterogeneous traffic conditions.

- The average MAPE and GEH values between the input and simulated measures for the intersections under study during calibration and validation are acceptable and conform to the limits of 15% and 5.0 respectively.

- The optimally best fit value of the sensitive parameters obtained were used to recommend a range of values for the calibration parameters which are shown below:

  i. Minimum Look Ahead Distance = [10 – 20] m
  ii. Minimum Look Back Distance = [15 – 18.71] m
  iii. Maximum Look Back Distance = [107.94 – 150] m
  iv. Average Standstill Distance = [0.3 – 1.5] m
  v. Additive Part of Safety Distance = [0.1 – 0.5]
  vi. Multiplicative Part of Safety Distance = [0 – 1]

vii.   Minimum Clearance (Front/Rear) = [0.49 – 0.73] m

viii.   Minimum Lateral Distance (Standing) at 0 km/h = [0.2 - 0.41] m

ix.   Minimum Lateral Distance (Driving) at 50 km/h = [0.6 - 0.9] m

Based on the comparison of the results of the calibration process with similar literatures, we can see that the calibration values adopted during studies in Kathmandu are within the recommended range of this study. We can also see that that the results of the analyzed parameters of heterogeneous traffic conditions within Kathmandu are quite similar to the results of Indian heterogeneous traffic conditions based on the calibration values.

5.2   Recommendations and Scope for Future Research

A methodology to model and calibrate heterogeneous traffic conditions using genetic algorithm has been presented in this study which was performed in VISSIM. Automated calibration process has greatly reduced the manual effort required for the calibration of models. This study is expected to help future practitioners in developing simulation models during calibration of VISSIM models. It is always recommended that a simulation model be calibrated before any analysis. This study recommends the use of the identified sensitive calibration parameters and their range of values during calibration of VISSIM at intersections for heterogeneous traffic conditions in Kathmandu. It is expected that the use of these sensitive calibration parameters and their ranges would significantly reduce the time and effort consumed during calibration of the VISSIM models. Addressing each and every facet of a subject within a restricted timeframe proves to be a difficult task. The subsequent tasks are suggested for further academic investigation.

1. Conducting similar research at corridor or network level is suggested.
2. Driver behavior may change when it comes to rural areas so further research could be done in such areas.
3. This study uses traffic volume and queue length as measures of effectiveness for calibration and validation so further research is recommended using additional MOEs for the study.

4. Only 12 calibration parameters were studied so further research is recommended using additional calibration parameters such as Wiedemann 99 parameters.

5. The verification of the range of values recommended for the sensitive calibration parameters through field measurement is difficult and subjected to different factors. Further research is recommended to explore such areas.

6. Vehicle characteristics such as axle configuration and turning radius may also be incorporated in further research.

7. Conducting the same research using other traffic flow simulation software is suggested.

REFERENCES

Acharya, A. and A. Marsani (2020). Prediction of Traffic Conflicts at Signalized Intersection: A Case Study of New Baneshwor Intersection, Pulchowk Campus.

ADB (2016). Kathmandu Sustainable Urban Transport Project (KSUTP).

Arnold, J. C. and J. S. Milton (2003). Introduction to probability and statistics: principles and applications for engineering and the computing sciences, McGraw-Hill.

Arroju, R., et al. (2015). "Comparative evaluation of roundabout capacities under heterogeneous traffic conditions." Journal of Modern Transportation **23**: 310-324.

Caliendo, C. (2014). "Delay time model at unsignalized intersections." Journal of transportation engineering **140**(9): 04014042.

Cunto, F. and F. F. Saccomanno (2008). "Calibration and validation of simulated vehicle safety performance at signalized intersections." Accident analysis & prevention **40**(3): 1171-1179.

Department of Roads, G. o. N. (2013). Nepal Road Standard.

Department of Transport Management, G. o. N. (2019). Vehicle Registration Statistic.

Dowling, R., et al. (2004). Traffic analysis toolbox, volume III: Guidelines for applying traffic microsimulation modeling software, United States. Federal Highway Administration. Office of Operations.

Dutta, M. and M. Ahmed (2019). "Calibration of VISSIM models at three-legged unsignalized intersections under mixed traffic conditions." Advances in Transportation Studies **48**.

FDOT (2014). Traffic Analysis Handbook, Florida Department of Transportaion.

Ge, Q. and M. Menendez (2012). Sensitivity analysis for calibrating VISSIM in modeling the Zurich network. 12th Swiss transport research conference.

Holland, J. H. (1992). Adaptation in natural and artificial systems: an introductory analysis with applications to biology, control, and artificial intelligence, MIT press.

Indiamart (2023). "Force Traveller 26 Seater Staff Bus." from https://dir.indiamart.com/items/force-traveller-26-seater-staff-bus-s106642.html.

Jayasooriya, N. and S. Bandara (2018). Calibrating and validating VISSIM microscopic simulation software for the context of Sri Lanka. 2018 Moratuwa Engineering Research Conference (MERCon), IEEE.

JICA, S. (2012). Data Collection Survey on Traffic Improvement in Kathmandu Valley. 2nd Workshop.[Presentation Slides].

Lownes, N. E. and R. B. Machemehl (2006). VISSIM: a multi-parameter sensitivity analysis. Proceedings of the 2006 Winter Simulation Conference, IEEE.

Ma, T. and B. Abdulhai (2002). "Genetic algorithm-based optimization approach and generic tool for calibrating traffic microscopic simulation parameters." Transportation research record **1800**(1): 6-15.

Manjunatha, P., et al. (2013). Methodology for the Calibration of VISSIM in Mixed Traffic. Transportation research board 92nd annual meeting, Transportation Research Board Washington, DC, United States.

Mathew, T. V. and P. Radhakrishnan (2010). "Calibration of microsimulation models for nonlane-based heterogeneous traffic at signalized intersections." Journal of Urban Planning and Development **136**(1): 59-66.

McKay, M. D., et al. (2000). "A comparison of three methods for selecting values of input variables in the analysis of output from a computer code." Technometrics **42**(1): 55-61.

Mondal, S. and A. Gupta (2020). Microsimulation-based framework to analyse urban signalised intersection in mixed traffic. Proceedings of the Institution of Civil Engineers-Transport, Thomas Telford Ltd.

Montsi, M. A. (2020). Calibration and validation of a micro-simulation model for operational performance evaluation of urban corridors, Cape Peninsula University of Technology.

ODOT (2011). Protocol for VISSIM Simulation. Oregon Department of Transportation.

Park, B. and H. Qi (2005). "Development and Evaluation of a Procedure for the Calibration of Simulation Models." Transportation research record **1934**(1): 208-217.

Park, B. and J. Schneeberger (2003). "Microscopic simulation model calibration and validation: case study of VISSIM simulation model for a coordinated actuated signal system." Transportation research record **1856**(1): 185-192.

Park, B., et al. (2006). "Application of microscopic simulation model calibration and validation procedure: Case study of coordinated actuated signal system." Transportation research record **1978**(1): 113-122.

Pattnaik, S., et al. (1998). "Urban bus transit route network design using genetic algorithm." Journal of transportation engineering **124**(4): 368-375.

Pradhananga, R., et al. (2011). "Optimisation model for hazardous material transport routing in Thailand." International journal of logistics systems and management **9**(1): 22-42.

PTV AG, K., Germany (2022). PTV VISSIM 2022 User Manual.

PTV AG, K., Germany (2023). "PTV VISSIM Academic License." from https://company.ptvgroup.com/en/about/academia.

Punzo, V. (2011). COST Action TU0903 Annual Progress Conference Report. COST TUD Progress Annual Conference, Vienna.

Road Safety and Traffic Unit, D. o. R., Government of Nepal (2023). Kathmandu Valley Intelligent Traffic Signal Project (KVITSP).

Road Safety and Traffic Unit, D. o. R., Government of Nepal (November, 2021). 72 hours traffic count for signal design of different intersections of Kathmandu Valley.

Rrecaj, A. A. and K. MBombol (2015). "Calibration and Validation of the VISSIM Parameters-State of the Art." TEM journal **4**(3): 255.

Sharma, A. K. (2016). Comparison of Probable Congestion Reduction Approaches at New Baneshwar Intersection in Kathmandu, Pokhara University.

Shrestha, A. (2022). Modelling Delay due to Curb-Side Bus Stops at Signalized Intersection: A Case Study of New Baneshwor Intersection. Nepal Engineering College, Changunarayan, Bhaktapur, Nepal, Pokhara University. **Master of Science in Transportation Engineering and Management**.

Shrestha, S. and A. Marsani (2017). Performance Improvement of a Signalized Intersection (A Case Study of New Baneshwor Intersection). Proceedings of IOE Graduate Conference.

Siddharth, S. P. and G. Ramadurai (2013). "Calibration of VISSIM for Indian heterogeneous traffic conditions." Procedia-Social and Behavioral Sciences **104**: 380-389.

Sun, D. J., et al. (2013). "Comparative study on simulation performances of CORSIM and VISSIM for urban street network." Simulation Modelling Practice and Theory **37**: 18-29.

Tettamanti, T., et al. (2015). "Iterative calibration of VISSIM simulator based on genetic algorithm." Acta Technica Jaurinensis **8**(2): 145-152.

Tianzi, C., et al. (2013). "Comparative study of VISSIM and SIDRA on signalized intersection." Procedia-Social and Behavioral Sciences **96**: 2004-2010.

VDOT (2020). "VDOT VISSIM User Guide—Version 2.0." Richmond, VA, USA.

Veerkamp, H. (2015). "Various buses in Nepal." from https://www.bus-planet.com/bus/bus-asia/Nepal-site/various-buses/index.html.

Zhizhou, W., et al. (2005). <u>Calibration of VISSIM for shanghai expressway using genetic algorithm</u>. Proceedings of the Winter Simulation Conference, 2005., IEEE.

APPENDICES

APPENDIX A Sample Data of Classified Traffic Volume Count

Sample Data of Maitighar Leg of Singhadurbar Intersection (Day 1)

		Station: Singhadurbar																										Month: August		
Leg:	Maitighar				a:	Sahidgate																			b:			Putalisadak		
Start Time	End Time	Truck				Bus								Car		Motor Cycle		Utility Vehicles		4Wheel Drive		Tractor		3-Wheeler		Power Triller		Bicycle		Total
		Multi axle		Heavy		Light		Big		Mini		Micro																		
		a	b	a	b	a	b	a	b	a	b	a	b	a	b	a	b	a	b	a	b	a	b	a	b	a	b	a	b	
Day 1																														
00:00:00	00:15:00	0	0	0	0	0	0	0	0	0	0	0	0	1	0	1	2	0	1	0	1	0	0	0	0	0	0	0	0	6
00:15:00	00:30:00	0	0	0	0	0	0	0	0	0	0	0	0	0	1	2	1	0	1	0	0	0	0	0	0	0	0	0	0	5
00:30:00	00:45:00	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	3	0	0	0	1	0	0	0	0	0	0	0	1	6
00:45:00	01:00:00	0	0	0	0	0	0	0	0	0	0	0	0	0	2	0	1	0	2	0	0	0	0	0	0	0	0	0	0	5
01:00:00	01:15:00	0	0	0	0	0	0	0	0	0	0	0	0	0	4	1	2	0	0	0	0	0	0	0	0	0	0	0	0	7
01:15:00	01:30:00	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	2	0	4	0	0	0	0	0	0	0	0	0	0	7
01:30:00	01:45:00	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	1	0	2	0	0	0	0	0	0	0	0	0	0	4
01:45:00	02:00:00	0	0	0	0	0	0	0	0	1	0	0	0	0	4	0	2	0	2	0	1	0	0	0	0	0	0	0	0	10
02:00:00	02:15:00	0	0	0	0	0	0	0	0	0	0	0	0	0	2	0	3	0	2	0	0	0	0	0	0	0	0	0	1	8
02:15:00	02:30:00	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	3	0	1	0	1	0	0	0	0	0	0	0	0	6
02:30:00	02:45:00	0	0	0	0	0	0	0	0	0	0	0	0	0	6	0	4	1	1	0	0	0	0	0	0	0	0	0	0	12
02:45:00	03:00:00	0	0	0	0	0	0	0	0	0	0	0	0	0	3	0	2	0	1	0	1	0	0	0	0	0	0	0	0	7
03:00:00	03:15:00	0	0	0	0	0	0	0	0	1	0	0	0	0	4	0	9	0	4	0	2	0	0	0	0	0	0	0	1	21
03:15:00	03:30:00	0	0	0	0	0	0	0	0	0	0	0	0	0	2	0	5	0	1	0	2	0	0	0	0	0	0	0	0	10
03:30:00	03:45:00	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	3	0	1	0	0	0	0	0	0	0	0	1	0	6
03:45:00	04:00:00	0	0	0	0	0	0	0	0	0	0	0	0	1	1	0	5	0	2	1	1	0	0	0	0	0	0	1	0	12
04:00:00	04:15:00	0	0	0	0	0	0	0	0	0	1	0	0	1	1	0	7	0	2	0	3	0	0	0	0	0	0	0	2	17
04:15:00	04:30:00	0	0	0	0	0	0	0	0	0	2	0	0	0	1	0	6	0	2	0	2	0	0	0	0	0	0	1	1	15
04:30:00	04:45:00	0	0	0	0	0	0	0	0	1	0	0	0	0	4	0	9	0	4	1	0	0	0	0	1	0	0	1	0	21
04:45:00	05:00:00	0	0	0	0	0	0	0	0	0	0	1	0	1	1	1	10	0	1	0	1	0	0	0	0	0	0	0	1	17
05:00:00	05:15:00	0	0	0	0	0	0	0	0	0	1	2	0	1	1	1	11	0	2	1	1	0	0	0	1	0	0	3	2	27
05:15:00	05:30:00	0	0	0	0	0	0	0	0	0	0	1	0	3	4	0	16	0	3	0	1	0	0	0	1	0	0	1	2	32
05:30:00	05:45:00	0	0	0	0	0	0	2	0	1	0	1	0	1	9	1	22	0	1	1	1	0	0	0	0	0	0	0	3	43
05:45:00	06:00:00	0	0	0	0	0	0	1	0	2	0	0	0	1	11	3	31	0	1	0	1	0	0	0	0	0	0	1	2	54
06:00:00	06:15:00	0	0	0	0	0	1	1	0	4	2	0	0	0	13	3	37	0	0	0	3	0	0	0	0	0	0	1	5	70
06:15:00	06:30:00	0	0	0	0	0	0	2	8	2	0	0	0	2	16	2	67	0	2	1	2	0	0	0	1	0	0	2	4	111
06:30:00	06:45:00	0	0	0	0	0	0	2	6	3	0	0	0	1	19	1	65	1	0	2	3	0	0	0	0	0	0	1	3	107
06:45:00	07:00:00	0	0	0	1	0	0	2	5	2	0	0	0	0	25	2	79	0	1	2	2	0	0	0	0	0	0	0	2	123
07:00:00	07:15:00	0	0	0	0	0	0	3	7	1	0	0	0	1	21	3	81	1	2	0	3	0	0	0	0	0	0	2	5	130
07:15:00	07:30:00	0	0	0	0	0	2	1	5	3	1	0	0	2	23	6	85	0	1	1	2	0	0	1	0	0	0	1	6	140
07:30:00	07:45:00	0	0	0	0	0	0	3	6	1	0	0	0	1	24	5	98	2	0	2	4	0	0	0	0	0	0	0	4	150
07:45:00	08:00:00	0	0	0	0	0	0	7	5	1	1	0	0	0	28	7	103	1	2	1	2	0	0	0	0	0	0	2	2	162

Sample Data of Singhadurbar Leg of Padmodaya Intersection (Day 2)

		Station: Padmodaya																Month: August											
Leg:	Singhadurbar					a:	Anamnagar												b:			Putalisadak								
		Truck				Bus								Car		Motor Cycle		Utility Vehicles		4Wheel Drive		Tractor		3-Wheeler		Power Triller		Bicycle		Total
Start Time	End Time	Multi axle		Heavy		Light		Big		Mini		Micro																		
		a	b	a	b	a	b	a	b	a	b	a	b	a	b	a	b	a	b	a	b	a	b	a	b	a	b	a	b	
Day 2																														
09:00:00	09:15:00	0	0	0	0	0	0	0	0	0	3	0	0	36	48	136	227	0	0	7	2	0	0	5	0	0	0	1	5	470
09:15:00	09:30:00	0	0	0	0	0	0	0	0	0	10	0	0	39	51	154	258	0	0	5	5	0	0	8	0	0	0	2	3	535
09:30:00	09:45:00	0	0	0	0	0	0	0	0	0	9	0	0	41	51	164	278	0	0	4	2	0	0	5	0	0	0	1	6	561
09:45:00	10:00:00	0	0	0	0	0	0	0	0	0	10	0	0	32	47	151	293	0	0	4	3	0	0	3	0	0	0	3	3	549
10:00:00	10:15:00	0	0	0	0	0	0	0	0	1	7	0	0	28	35	154	303	0	0	6	5	0	0	8	0	0	0	1	6	554
10:15:00	10:30:00	0	0	0	0	0	0	0	0	0	12	0	0	31	33	165	272	0	0	3	9	0	0	3	0	0	0	2	4	534
10:30:00	10:45:00	0	0	0	0	0	0	0	0	0	11	0	0	35	27	162	263	0	0	1	8	0	0	4	0	0	0	0	5	516
10:45:00	11:00:00	0	0	0	0	0	0	0	0	0	11	0	0	45	34	186	250	0	0	7	10	0	0	7	0	0	0	0	7	557
11:00:00	11:15:00	0	0	0	0	0	0	0	0	2	10	0	0	47	32	194	249	0	0	5	12	0	0	3	0	0	0	3	2	559
11:15:00	11:30:00	0	0	0	0	0	0	0	0	0	7	0	0	53	30	173	256	0	0	11	11	0	0	5	0	0	0	2	6	554
11:30:00	11:45:00	0	0	0	0	0	0	0	0	0	12	0	0	50	28	171	260	0	0	10	10	0	0	4	0	0	0	2	4	551
11:45:00	12:00:00	0	0	0	0	0	0	0	0	0	6	0	0	32	18	160	251	0	0	9	6	0	0	6	0	0	0	0	3	491
12:00:00	12:15:00	0	0	0	0	0	0	0	0	0	7	0	0	48	29	146	249	0	2	15	12	0	0	9	0	0	0	2	3	522
12:15:00	12:30:00	0	0	0	0	0	0	0	0	0	4	0	0	58	33	168	246	0	0	13	7	0	0	7	0	0	0	3	4	543
12:30:00	12:45:00	0	0	0	0	0	0	0	0	0	6	0	0	67	31	156	238	0	0	11	8	0	0	8	0	0	0	2	6	533
12:45:00	13:00:00	0	0	0	0	0	0	0	0	0	5	0	0	53	27	140	221	0	0	9	8	0	0	4	0	0	0	1	5	473
13:00:00	13:15:00	0	0	0	0	0	0	0	0	2	8	0	0	69	43	151	212	0	0	3	9	0	0	8	0	0	0	0	3	508
13:15:00	13:30:00	0	0	0	0	0	0	0	0	0	4	0	0	51	24	132	216	0	0	6	5	0	0	3	0	0	0	0	3	444
13:30:00	13:45:00	0	0	0	0	0	0	0	0	0	2	0	0	59	29	128	209	0	0	9	8	0	0	8	0	0	0	3	3	458
13:45:00	14:00:00	0	0	0	0	0	0	0	0	0	3	0	0	50	27	135	223	0	0	6	6	0	0	3	0	0	0	1	3	457
14:00:00	14:15:00	0	0	0	0	0	0	0	0	0	7	0	0	57	34	142	223	0	0	10	8	0	0	2	0	0	0	5	1	489
14:15:00	14:30:00	0	0	0	0	0	0	0	0	0	7	0	0	70	30	153	236	0	0	10	6	0	0	4	0	0	0	5	2	523
14:30:00	14:45:00	0	0	0	0	0	0	0	0	0	4	0	0	47	25	144	220	0	0	9	6	0	0	4	0	0	0	3	3	465
14:45:00	15:00:00	0	0	0	0	0	0	0	0	0	4	0	0	51	27	126	219	0	0	12	5	0	0	6	0	0	0	2	2	454
15:00:00	15:15:00	0	0	0	0	0	0	0	0	0	4	0	0	36	18	142	240	0	0	5	6	0	0	1	0	0	0	2	3	457
15:15:00	15:30:00	0	0	0	0	0	0	0	0	0	3	0	0	37	32	146	251	0	0	16	4	0	0	4	0	0	0	3	3	499
15:30:00	15:45:00	0	0	0	0	0	0	0	0	2	4	0	0	48	36	154	239	0	0	9	7	0	0	3	0	0	0	3	2	507
15:45:00	16:00:00	0	0	0	0	0	0	0	0	0	8	0	0	40	37	131	242	0	0	4	2	0	0	2	0	0	0	3	2	471
16:00:00	16:15:00	0	0	0	0	0	0	0	0	0	9	0	0	33	41	142	226	0	0	7	6	0	0	5	0	0	0	2	0	471
16:15:00	16:30:00	0	0	0	0	0	0	0	0	0	6	0	0	25	30	153	238	0	0	2	8	0	0	3	0	0	0	4	4	472
16:30:00	16:45:00	0	0	0	0	0	0	0	0	0	5	0	0	28	44	163	263	0	0	4	5	0	0	7	0	0	0	4	2	525
16:45:00	17:00:00	0	0	0	0	0	0	0	0	0	10	0	0	45	35	194	278	0	0	10	3	0	0	2	0	0	0	2	4	583
17:00:00	17:15:00	0	0	0	0	0	0	0	0	0	3	0	0	37	25	156	289	0	0	6	8	0	0	3	0	0	0	2	2	531
17:15:00	17:30:00	0	0	0	0	0	0	0	0	0	2	0	0	44	40	200	303	0	0	10	8	0	0	2	0	0	0	4	2	615
17:30:00	17:45:00	0	0	0	0	0	0	0	0	0	4	0	0	38	45	203	312	0	0	12	8	0	0	4	0	0	0	2	3	631
17:45:00	18:00:00	0	0	0	0	0	0	0	0	0	4	0	0	38	35	186	277	0	0	5	4	0	0	2	0	0	0	2	4	557

Sample Data of Bagbazar Leg of Putalisadak Intersection (Day 3)

Station: Putalisadak Month: August

Leg: Bagbazar   a: Dillibazar   b: Hattisar   c: Singhadurbar

Start Time	End Time	Truck Multi axle a	b	c	Truck Heavy a	b	c	Bus Light a	b	c	Bus Big a	b	c	Bus Mini a	b	c	Bus Micro a	b	c	Car a	b	c	Motor Cycle a	b	c	Utility Vehicles a	b	c	4Wheel Drive a	b	c	Tractor a	b	c	3-Wheeler a	b	c	Power Triller a	b	c	Bicycle a	b	c	Total
Day 3																																											
17:00:00	17:15:00	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	24	0	0	177	0	0	0	0	0	7	0	0	0	0	0	0	0	0	0	0	0	1	6	0	215
17:15:00	17:30:00	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	25	0	0	182	0	0	0	0	0	9	0	0	0	0	0	0	0	0	0	0	0	3	10	1	230
17:30:00	17:45:00	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	19	0	0	169	0	0	2	0	0	10	0	0	0	0	0	0	0	0	0	0	0	3	3	0	206
17:45:00	18:00:00	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	15	0	0	155	1	0	1	0	0	7	0	0	0	0	0	0	0	0	0	0	0	2	4	1	186
18:00:00	18:15:00	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	23	0	0	195	0	0	0	0	0	11	0	0	0	0	0	0	0	0	0	0	0	4	4	0	237
18:15:00	18:30:00	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	21	0	0	197	0	0	0	0	0	8	0	0	0	0	0	0	0	0	0	0	0	7	10	0	243
18:30:00	18:45:00	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	29	0	0	190	0	0	0	0	0	10	0	0	0	0	0	0	0	0	0	0	0	3	4	0	236
18:45:00	19:00:00	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	27	0	0	179	0	0	0	0	0	13	0	0	0	0	0	0	0	0	0	0	0	2	6	0	227
19:00:00	19:15:00	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	32	0	0	192	0	0	2	0	0	11	0	0	0	0	0	0	0	0	0	0	0	4	10	2	253
19:15:00	19:30:00	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	23	0	0	187	0	0	2	0	0	15	0	0	0	0	0	0	0	0	0	0	0	5	13	0	245
19:30:00	19:45:00	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	25	0	0	172	0	0	0	0	0	13	0	0	0	0	0	0	0	0	0	0	0	3	9	2	224
19:45:00	20:00:00	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	21	0	0	168	0	0	2	0	0	12	0	0	0	0	0	0	0	0	0	0	0	6	8	0	217
20:00:00	20:15:00	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	15	0	0	130	0	0	1	0	0	9	0	0	0	0	0	0	0	0	1	0	0	2	5	2	165
20:15:00	20:30:00	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	10	0	0	126	0	0	1	0	0	8	0	0	0	0	0	0	0	0	0	0	0	1	1	0	147
20:30:00	20:45:00	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	8	0	0	131	0	0	2	0	0	9	0	0	0	0	0	0	0	0	0	0	0	1	0	1	152
20:45:00	21:00:00	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	9	0	0	90	0	0	0	0	0	6	0	0	0	0	0	0	0	0	1	0	0	1	2	1	110
21:00:00	21:15:00	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	6	0	0	73	0	0	0	0	0	7	0	0	0	0	0	0	0	0	0	0	0	2	0	2	91
21:15:00	21:30:00	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	3	0	0	66	0	0	0	0	0	6	0	0	0	0	0	0	0	0	0	0	0	3	0	0	78
21:30:00	21:45:00	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	2	0	0	55	0	0	0	0	0	5	0	0	0	0	0	0	0	0	1	0	0	2	1	0	66
21:45:00	22:00:00	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	42	0	0	0	1	0	4	0	0	0	0	0	0	0	0	0	0	0	1	0	0	49
22:00:00	22:15:00	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	2	0	0	32	0	0	0	0	0	8	0	0	0	0	0	0	0	0	0	0	0	2	0	0	44
22:15:00	22:30:00	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	21	0	0	0	0	1	6	0	0	0	0	0	0	0	0	0	0	0	1	0	0	29
22:30:00	22:45:00	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	2	0	0	19	0	0	0	0	0	5	0	0	0	0	0	0	0	0	0	0	0	0	1	0	27
22:45:00	23:00:00	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	15	0	0	0	0	0	2	0	0	0	0	0	0	0	0	1	0	0	1	0	0	19
23:00:00	23:15:00	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	10	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	10
23:15:00	23:30:00	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	6	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	8
23:30:00	23:45:00	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	8	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	8
23:45:00	00:00:00	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	9	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	10
Total		0	0	0	0	1	0	0	0	0	0	0	0	1	0	0	0	0	2	1020	2	2	8377	33	3	65	3	0	432	0	0	0	0	0	0	0	0	7	0	0	157	249	68	**10422**

APPENDIX B Traffic Volume Count Summary

24 hour Traffic Volume Summary of Singhadurbar Intersection

Day	24 hour Traffic Volume Count Summary								
	M-S	M-P	P-S	P-M	S-P	S-M	Total Vehicles	Total PCU	Average Daily Traffic
Day 1	5043	17271	4246	17921	5750	37058	**87289**	**47487.5**	**91417.67**
Day 2	5074	18698	4874	20212	6574	41490	**96922**	**52921.6**	
Day 3	5406	17827	4417	18430	5942	38020	**90042**	**49449.7**	
Similar to the above tables, M = Maitighar, S = Sahidgate, and P = Putalisadak									

24 hour Traffic Volume Summary of Putalisadak Intersection

Day	24 hour Traffic Volume Count Summary														
	B-D	B-H	B-S	D-B	D-H	D-S	S-D	S-B	S-H	H-D	H-B	H-S	Total Vehicles	Total PCU	Average Daily Traffic
Day 1	130	9496	78	7451	7344	9702	22	6548	20783	36	23	25867	**87480**	**41406.2**	**93478**
Day 2	288	11158	323	8660	8856	11316	34	7872	20695	62	34	30637	**99935**	**48741.7**	
Day 3	163	10146	113	7959	8106	10437	18	7400	20870	56	26	27725	**93019**	**45272.2**	
Similar to the above tables, B = Bagbazar, D = Dillibazar, S = Singhadurbar, and H = Hattisar															

24 hour Traffic Volume Summary of Padmodaya Intersection

Day	24 hour Traffic Volume Count Summary								
	S-A	S-P	P-A	P-S	A-S	A-P	Total Vehicles	Total PCU	Average Daily Traffic
Day 1	9341	14651	13059	16955	10798	14188	**78992**	**36670.6**	**80736.67**
Day 2	10173	14722	12232	18011	12429	15178	**82745**	**39494.3**	
Day 3	10153	13143	11953	18159	11578	15487	**80473**	**38705.3**	
Similar to the above tables, A = Anamnagar, S = Singhadurbar, and P = Putalisadak									

107

APPENDIX C Vehicle Routing and Relative Flow

Relative Flow in Approach Legs of Singhadurbar Intersection

Relative Flow in Singhadurbar Intersection (Maitighar to Respective Legs)

	Day 1	Day 2	Day 3
M-S	5043	5074	5406
M-P	17271	18698	17827

Relative Flow in Singhadurbar Intersection (Putalisadak to Respective Legs)

	Day 1	Day 2	Day 3
P-S	4246	4874	4417
P-M	17921	20212	18430

Relative Flow in Singhadurbar Intersection (Sahidgate to Respective Legs)

	Day 1	Day 2	Day 3
S-P	5750	6574	5942
S-M	37058	41490	38020

Relative Flow in Approach Legs of Putalisadak Intersection

Relative Flow in Putalisadak Intersection (Bagbazar to Respective Legs)

	Day 1	Day 2	Day 3
B-D	130	288	163
B-H	9496	11158	10146
B-S	78	323	113

Relative Flow in Putalisadak Intersection (Dillibazar to Respective Legs)

	Day 1	Day 2	Day 3
D-B	7451	8660	7959
D-H	7344	8856	8106
D-S	9702	11316	10437

109

Relative Flow in Approach Legs of Putalisadak Intersection (contd.)

Relative Flow in Putalisadak Intersection
(Singhadurbar to Respective Legs)

	Day 1	Day 2	Day 3
S-D	22	34	18
S-B	6548	7872	7400
S-H	20783	20695	20870

S-D S-B S-H

Relative Flow in Putalisadak Intersection
(Hattisar to Respective Legs)

	Day 1	Day 2	Day 3
H-D	36	62	56
H-B	23	34	26
H-S	25867	30637	27725

H-D H-B H-S

Relative Flow in Approach Legs of Padmodaya Intersection

Relative Flow in Padmodaya Intersection (Singhadurbar to Respective Legs)

	Day 1	Day 2	Day 3
S-A	9341	10173	10153
S-P	14651	14722	13143

Relative Flow in Padmodaya Intersection (Putalisadak to Respective Legs)

	Day 1	Day 2	Day 3
P-A	13059	12232	11953
P-S	16955	18011	18159

Relative Flow in Padmodaya Intersection (Anamnagar to Respective Legs)

	Day 1	Day 2	Day 3
A-S	10798	12429	11578
A-P	14188	15178	15487

APPENDIX D Samples of SPSS Output for First Level and Second Level One Way ANOVA Testing

Output
- Log
- Oneway
  - Title
  - Notes
  - Active Dataset
  - Descriptives
  - Test of Homogen
  - ANOVA
  - Robust Tests of E
  - Means Plots
    - Title
    - Sahidgate_Tr
    - Maitighar_Tri
    - Putalisadak_
- Log
- Oneway
  - Title
  - Notes
  - Descriptives
  - Test of Homogen
  - ANOVA
  - Robust Tests of E
  - Means Plots
    - Title
    - Sahidgate_Tr

		Statistic	df1	df2	Sig.
	Based on Median and with adjusted df	1.375	4	168.450	.245
	Based on trimmed mean	1.914	4	195	.110

ANOVA

		Sum of Squares	df	Mean Square	F	Sig.
Sahidgate_Trial1	Between Groups	23.365	4	5.841	1.340	.257
	Within Groups	850.322	195	4.361		
	Total	873.686	199			
Maitighar_Trial2	Between Groups	45.396	4	11.349	1.293	.274
	Within Groups	1712.056	195	8.780		
	Total	1757.453	199			
Putalisadak_Trial3	Between Groups	22.245	4	5.561	1.149	.335
	Within Groups	944.158	195	4.842		
	Total	966.402	199			

Robust Tests of Equality of Means

		Statistic[a]	df1	df2	Sig.
Sahidgate_Trial1	Welch	1.273	4	96.434	.286
	Brown-Forsythe	1.340	4	165.293	.257
Maitighar_Trial2	Welch	1.205	4	97.234	.314
	Brown-Forsythe	1.293	4	185.353	.275
Putalisadak_Trial3	Welch	1.018	4	96.385	.402
	Brown-Forsythe	1.149	4	169.252	.336

a. Asymptotically F distributed.

Means Plots

6.600000

Output
  Log
  Oneway
    Title
    Notes
    Active Dataset
    Descriptives
    Test of Homogene
    ANOVA
    Robust Tests of E
    Means Plots
      Title
      Sahidgate_Tr
      Maitighar_Tri
      Putalisadak_
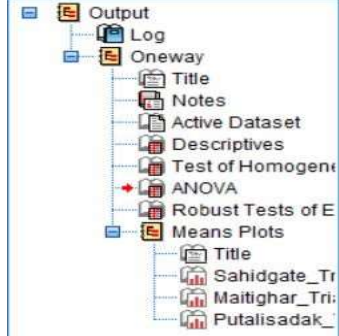
			df1		Sig.
Based on Median		29.291	4	195	.000
Based on Median and with adjusted df		29.291	4	66.179	.000
Based on trimmed mean		41.617	4	195	.000

ANOVA

		Sum of Squares	df	Mean Square	F	Sig.
Sahidgate_Trail1	Between Groups	462.601	4	115.650	54.850	.000
	Within Groups	411.151	195	2.108		
	Total	873.752	199			
Maitighar_Trial2	Between Groups	551.262	4	137.816	22.277	.000
	Within Groups	1206.365	195	6.186		
	Total	1757.627	199			
Putalisadak_Trial3	Between Groups	595.860	4	148.965	78.399	.000
	Within Groups	370.519	195	1.900		
	Total	966.379	199			

Robust Tests of Equality of Means

		Statistic[a]	df1	df2	Sig.
Sahidgate_Trail1	Welch	38.510	4	94.749	.000
	Brown-Forsythe	54.850	4	77.077	.000
Maitighar_Trial2	Welch	20.085	4	90.645	.000
	Brown-Forsythe	22.277	4	124.103	.000
Putalisadak_Trial3	Welch	51.857	4	94.193	.000
	Brown-Forsythe	78.399	4	67.693	.000

a. Asymptotically F distributed.

Means Plots

113
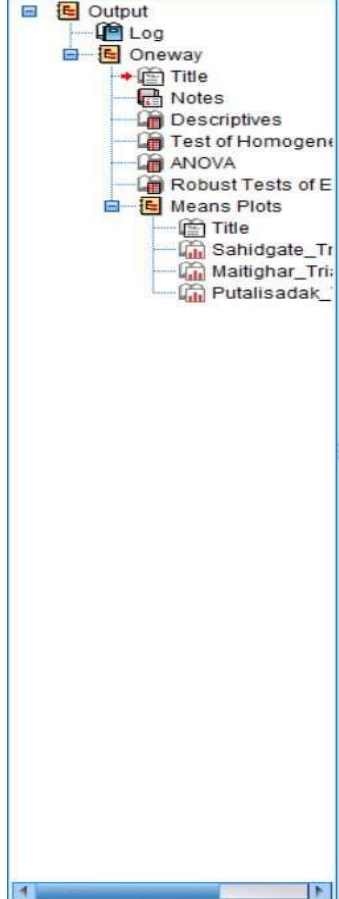
Putalisadak_Trial3	Based on Mean	21.758	2	197	.000
	Based on Median	9.492	2	197	.000
	Based on Median and with adjusted df	9.492	2	116.326	.000
	Based on trimmed mean	18.943	2	197	.000

ANOVA

		Sum of Squares	df	Mean Square	F	Sig.
Sahidgate_Trial1	Between Groups	102.479	2	51.240	13.088	.000
	Within Groups	771.273	197	3.915		
	Total	873.752	199			
Maitighar_Trial2	Between Groups	149.797	2	74.899	9.177	.000
	Within Groups	1607.830	197	8.162		
	Total	1757.627	199			
Putalisadak_Trial3	Between Groups	81.102	2	40.551	9.024	.000
	Within Groups	885.276	197	4.494		
	Total	966.379	199			

Robust Tests of Equality of Means

		Statistic[a]	df1	df2	Sig.
Sahidgate_Trial1	Welch	9.188	2	123.556	.000
	Brown-Forsythe	13.133	2	129.129	.000
Maitighar_Trial2	Welch	7.314	2	126.049	.001
	Brown-Forsythe	9.193	2	159.370	.000
Putalisadak_Trial3	Welch	6.396	2	122.275	.002
	Brown-Forsythe	9.052	2	129.405	.000

a. Asymptotically F distributed.

Means Plots

114

File   Edit   View   Data   Transform   Insert   Format   Analyze   Graphs   Utilities   Extensions   Window   Help

- Output
  - Log
  - Oneway
    - Title
    - Notes
    - Descriptives
    - Test of Homogene
    - ANOVA
    - Robust Tests of E
    - Means Plots
      - Title
      - Sahidgate_Tr
      - Maitighar_Tri
      - Putalisadak_

Based on trimmed mean		1.632	4	195	.168

ANOVA

		Sum of Squares	df	Mean Square	F	Sig.
Sahidgate_Trial1	Between Groups	1.022	4	.256	.298	.879
	Within Groups	166.958	195	.856		
	Total	167.981	199			
Maitighar_Trial2	Between Groups	1.486	4	.371	.763	.550
	Within Groups	94.913	195	.487		
	Total	96.398	199			
Putalisadak_Trial3	Between Groups	.829	4	.207	.756	.555
	Within Groups	53.502	195	.274		
	Total	54.331	199			

Robust Tests of Equality of Means

		Statistic[a]	df1	df2	Sig.
Sahidgate_Trial1	Welch	.280	4	97.426	.890
	Brown-Forsythe	.298	4	192.511	.879
Maitighar_Trial2	Welch	.856	4	97.329	.493
	Brown-Forsythe	.763	4	187.057	.551
Putalisadak_Trial3	Welch	.642	4	97.189	.634
	Brown-Forsythe	.756	4	179.839	.555

a. Asymptotically F distributed.

Means Plots

1.350000

115

APPENDIX E Sample of Python Programming Code for First Level Sensitivity Analysis

```
# Sensitivity Analysis of VISSIM Parameters for Modeling Heterogeneous Traffic Conditions Using
Latin Hypercube Sampling.
# VISSIM Parameter Sensitivity Analysis
# Latin Hypercube Sampling to reduce range

# Authors: Aashish Manandhar, Buddha Thapa Magar

import numpy as np
import matplotlib.pyplot as plt
import win32com.client as com # VISSIM COM
from pyDOE import *
from scipy.stats.distributions import norm
import pandas as pd

if __name__ == "__main__":
    LookAheadDistMax = [200, 250, 300, 350];
    LookAheadDistMin = [10, 15, 20, 25, 30];
    LookBackDistMin = [5, 10, 15, 20, 25, 30];
    LookBackDistMax = [80, 120, 150, 180];
    W74ax = [0.3, 0.7, 1, 1.5, 2];
    W74bxAdd = [0.1, 0.5, 1, 1.5, 2];
    W74bxMult = [0, 1, 2, 3];
    DiffusTm = [30, 50, 75];
    MinFrontRearClear = [0.1, 0.5, 1];
    SafDistFactLnChg = [0.2, 0.4, 0.6, 0.7];
    LatDistStandDef = [0.1, 0.2, 0.5];
    LatDistDrivDef = [0.6, 0.8, 1];

    Pairs = [LookAheadDistMax, LookAheadDistMin, LookBackDistMin, LookBackDistMax, W74ax,
W74bxAdd, W74bxMult, DiffusTm, MinFrontRearClear, SafDistFactLnChg, LatDistStandDef,
LatDistDrivDef];
    PairsLength = [len(LookAheadDistMax), len(LookAheadDistMin), len(LookBackDistMin),
len(LookBackDistMax), len(W74ax), len(W74bxAdd), len(W74bxMult), len(DiffusTm),
len(MinFrontRearClear), len(SafDistFactLnChg), len(LatDistStandDef), len(LatDistDrivDef)];
#================================================================================
==========
# Latin Hypercube Sampling Design
# Candidate sets
    sam = 200 # sample size
    LHsets = lhs(12, samples = sam)                              # Generate candidate sets
    LHsets = norm(loc=0, scale=1).ppf(LHsets)      # Normalize the value of candidate sets to N(0,1)
    for i in range(12):                      # Substitute the LH matrix to real value
        Range = Pairs[i]
        N = PairsLength[i]
        Prob = 1/N
        Interval = [];
        for k in range(N-1):
            Interval.append(norm.ppf(Prob*(k+1)))
        for j in range(sam):
            for q in range(N-1):
                if Interval[q] > LHsets[j,i]:
                    LHsets[j,i] = Range[q]
                    break
```

```
            if q == N-2:
                LHsets[j,i] = Range[N-1]

    columns = ['LookAheadDistMax', 'LookAheadDistMin', 'LookBackDistMin', 'LookBackDistMax',
'W74ax', 'W74bxAdd', 'W74bxMult', 'DiffusTm', 'MinFrontRearClear', 'SafDistFactLnChg',
'LatDistStandDef', 'LatDistDrivDef'];

    parameterData = pd.DataFrame(LHsets, columns= columns);
    parameterData.to_csv("D:\SimulationOutput\parameterData.csv", header= columns);

#===============================================================================
=======
# VISSIM Configurations
    # Load VISSIM Network
    Vissim = com.Dispatch("Vissim.Vissim");
    Vissim.LoadNet(r'D:\Engineering\Master of Science in Transportation Engineering\4th Sem -
Thesis\03. Thesis\02. VISSIM Analysis\03. Singhadurbar - Latest\Singhadurbar Latest.inpx');
    # Define Simulation Configurations
    Sim = Vissim.Simulation;
    Net = Vissim.Net;
    G = Vissim.Graphics;
    dbpss = Net.DrivingBehaviors;                          # Driving behavior module
    dbps = dbpss.ItemByKey(1);                             # Urban (motorized)

    # Set Simulation Parameters
    TotalPeriod = 3900;        # Define total simulation period
    WarmPeriod = 300;          # Define warm period 5 minutes
    Random_Seed = 36;           # Define random seed
    Random_Seed_increment = 2
    step_time = 10             # Define Step Time
    total_number_of_stem_in_each_simulation = TotalPeriod * step_time
    Sim.SetAttValue('SimPeriod',TotalPeriod);
    Sim.SetAttValue('SimRes',step_time);
    Sim.SetAttValue('RandSeed', Random_Seed);
    Sim.SetAttValue('RandSeedIncr', Random_Seed_increment);
    Sim.SetAttValue('NumRuns',5);
    G.CurrentNetworkWindow.SetAttValue('QuickMode',1);
#===============================================================================
=======
# Test all candidate sets, to reduce the boundary

    output = []
    #Run = 1;
    for i in range(sam):
        dbps.SetAttValue('LookAheadDistMax',LHsets[i,0]);
        dbps.SetAttValue('LookAheadDistMin', LHsets[i,1]);
        dbps.SetAttValue('LookBackDistMin', LHsets[i,2]);
        dbps.SetAttValue('LookBackDistMax', LHsets[i,3]);
        dbps.SetAttValue('W74ax', LHsets[i,4]);
        dbps.SetAttValue('W74bxAdd', LHsets[i,5]);
        dbps.SetAttValue('W74bxMult', LHsets[i,6]);
        dbps.SetAttValue('DiffusTm', LHsets[i,7]);
        dbps.SetAttValue('MinFrontRearClear', LHsets[i,8]);
        dbps.SetAttValue('SafDistFactLnChg', LHsets[i,9]);
        dbps.SetAttValue('LatDistStandDef', LHsets[i,10]);
        dbps.SetAttValue('LatDistDrivDef', LHsets[i,11]);

        for runOfSimulation in range(1,6):
            for simulationStep in range(total_number_of_stem_in_each_simulation):
```

```
        if (round(Sim.AttValue("SimSec"),1) == 3899.9)  :                #To read output at the
end of a run of simulation
            for dataCollectionMeasurementKey in range(1,Net.DataCollectionMeasurements.Count
+1):
                dataCollectionMeasurement =
Net.DataCollectionMeasurements.ItemByKey(dataCollectionMeasurementKey)
                for timeInt in range(1,5):
                    attribute = 'Vehs(Current,'+ str(timeInt) + ', All)'
                    data = [i, timeInt, runOfSimulation, dataCollectionMeasurement.AttValue("Name"),
dataCollectionMeasurement.AttValue(attribute)]
                    output.append(data)
        Sim.RunSingleStep()
    Sim.Stop()
    columns = ['item','TimeInt', 'RunOfSimulation', 'Name', 'Volume']
    parameterData = pd.DataFrame(output, columns= columns)
    parameterData.to_csv("D:\SimulationOutput\SensitiveAnalysisOutput.csv", header= columns)
```

Appendix F Sample of Python Programming Code for Second Level Sensitivity Analysis

```python
# Sensitivity Analysis of VISSIM Parameters for Modeling Heterogeneous Traffic Conditions Using
Latin Hypercube Sampling (Second Order).
# VISSIM Parameter Sensitivity Analysis
# Latin Hypercube Sampling to reduce range

# Authors: Aashish Manandhar, Buddha Thapa Magar

import numpy as np
import matplotlib.pyplot as plt
import win32com.client as com # VISSIM COM
from pyDOE import *
from scipy.stats.distributions import norm
import pandas as pd

if __name__ == "__main__":
    LookAheadDistMax = [200, 250, 300, 350];
    LookAheadDistMin = [10, 15, 20, 25, 30];
    LookBackDistMax = [80, 120, 150, 180];
    DiffusTm = [30, 50, 60, 75];
    SafDistFactLnChg = [0.2, 0.35, 0.5, 0.6, 0.7];
    LatDistDrivDef = [0.6, 0.7, 0.8, 0.9, 1];

    Pairs = [LookAheadDistMax, LookAheadDistMin, LookBackDistMax, DiffusTm,
SafDistFactLnChg, LatDistDrivDef];
    PairsLength = [len(LookAheadDistMax), len(LookAheadDistMin), len(LookBackDistMax),
len(DiffusTm), len(SafDistFactLnChg), len(LatDistDrivDef)];
#================================================================================
========
# Latin Hypercube Sampling Design
# Candidate sets
    sam = 200 # sample size
    LHsets = lhs(6, samples = sam)                        # Generate candidate sets
    LHsets = norm(loc=0, scale=1).ppf(LHsets)      # Normalize the value of candidate sets to N(0,1)
    for i in range(6):                      # Substitute the LH matrix to real value
        Range = Pairs[i]
        N = PairsLength[i]
        Prob = 1/N
        Interval = [];
        for k in range(N-1):
            Interval.append(norm.ppf(Prob*(k+1)))
        for j in range(sam):
            for q in range(N-1):
                if Interval[q] > LHsets[j,i]:
                    LHsets[j,i] = Range[q]
                    break
                if q == N-2:
                    LHsets[j,i] = Range[N-1]

    columns = ['LookAheadDistMax', 'LookAheadDistMin', 'LookBackDistMax', 'DiffusTm',
'SafDistFactLnChg', 'LatDistDrivDef'];

    parameterData = pd.DataFrame(LHsets, columns= columns);
    parameterData.to_csv("D:\SecondOrderSimulationOutput\parameterData.csv", header= columns);
```

119

```
#===========================================================================
=======
# VISSIM Configurations
    # Load VISSIM Network
    Vissim = com.Dispatch("Vissim.Vissim");
    Vissim.LoadNet(r'D:\Engineering\Master of Science in Transportation Engineering\4th Sem -
Thesis\03. Thesis\02. VISSIM Analysis\03. Singhadurbar - Latest - SeconDOrder\Singhadurbar
Second Order.inpx');
    # Define Simulation Configurations
    Sim = Vissim.Simulation;
    Net = Vissim.Net;
    G = Vissim.Graphics;
    dbpss = Net.DrivingBehaviors;                                    # Driving behavior module
    dbps = dbpss.ItemByKey(1);                                       # Urban (motorized)

    # Set Simulation Parameters
    TotalPeriod = 3900;          # Define total simulation period
    WarmPeriod = 300;            # Define warm period 5 minutes
    Random_Seed = 36;             # Define random seed
    Random_Seed_increment = 2
    step_time = 10            # Define Step Time
    total_number_of_stem_in_each_simulation = TotalPeriod * step_time
    Sim.SetAttValue('SimPeriod',TotalPeriod);
    Sim.SetAttValue('SimRes',step_time);
    Sim.SetAttValue('RandSeed', Random_Seed);
    Sim.SetAttValue('RandSeedIncr', Random_Seed_increment);
    Sim.SetAttValue('NumRuns',5);
    G.CurrentNetworkWindow.SetAttValue('QuickMode',1);
#===========================================================================
=======
# Test all candidate sets, to reduce the boundary

    output = []
    #Run = 1;
    for i in range(sam):
        dbps.SetAttValue('LookAheadDistMax',LHsets[i,0]);
        dbps.SetAttValue('LookAheadDistMin', LHsets[i,1]);
        dbps.SetAttValue('LookBackDistMax', LHsets[i,2]);
        dbps.SetAttValue('DiffusTm', LHsets[i,3]);
        dbps.SetAttValue('SafDistFactLnChg', LHsets[i,4]);
        dbps.SetAttValue('LatDistDrivDef', LHsets[i,5]);

        for runOfSimulation in range(1,6):
            for simulationStep in range(total_number_of_stem_in_each_simulation):
                if (round(Sim.AttValue("SimSec"),1) == 3899.9)  :                     #To read output at the
end of a run of simulation
                    for dataCollectionMeasurementKey in range(1,Net.DataCollectionMeasurements.Count
+1):
                        dataCollectionMeasurement =
Net.DataCollectionMeasurements.ItemByKey(dataCollectionMeasurementKey)
                        for timeInt in range(1,5):
                            attribute = 'Vehs(Current,'+ str(timeInt) + ', All)'
                            data = [i, timeInt, runOfSimulation, dataCollectionMeasurement.AttValue("Name"),
dataCollectionMeasurement.AttValue(attribute)]
                            output.append(data)
                Sim.RunSingleStep()
        Sim.Stop()
    columns = ['item','TimeInt', 'RunOfSimulation', 'Name', 'Volume']
    parameterData = pd.DataFrame(output, columns= columns)
```

```
parameterData.to_csv("D:\SecondOrderSimulationOutput\SensitiveAnalysisOutput.csv", header=
columns)
```

APPENDIX G Sample of Python Programming Code for Automated Calibration Using Genetic Algorithm

```python
# Automated calibration of VISSIM parameters for modeling heterogeneous traffic conditions in
Kathmandu using Latin Hypercube Sampling and Genetic Algorithm
# Latin Hypercube Sampling to reduce range

# Authors: Aashish Manandhar, Buddha Thapa Magar

import numpy as np
import matplotlib.pyplot as plt
import os
import win32com.client as com # VISSIM COM
import geatpy as ga          # Genetic Algorithm Package
import time
from pyDOE import *
from scipy.stats.distributions import norm
import pandas as pd
import pygad

# Fitness Calculation for GA
def fitnessFunction(ga_instance, solution, solution_idx):
    print("Solution: ", solution)
    print("solution index: ", solution_idx)
    output = []
    queueLengthOutput = []
    dbps.SetAttValue('LookBackDistMax',solution[0])
    dbps.SetAttValue('LookAheadDistMin', solution[1])
    dbps.SetAttValue('LookBackDistMin', solution[2])
    dbps.SetAttValue('W74ax', solution[3])
    dbps.SetAttValue('W74bxAdd', solution[4])
    dbps.SetAttValue('W74bxMult', solution[5])
    dbps.SetAttValue('MinFrontRearClear', solution[6])
    dbps.SetAttValue('LatDistStandDef', solution[7])
    dbps.SetAttValue('LatDistDrivDef', solution[8])
    Sim.RunContinuous()
    for timeInt in range(1,5):

        # Queue length at three points:
        queueCounterPoints = Net.QueueCounters.GetMultiAttValues("Name")
        '''
        1: Sahidgate
        2: Putalisadak
        3: Maitighar
        '''

        for queueCounterPoint in queueCounterPoints:
            print(queueCounterPoint)
            queueCounterPointDetail = Net.QueueCounters.ItemByKey(queueCounterPoint[0])
            queueLengthData = [timeInt, queueCounterPoint[0],
queueCounterPointDetail.AttValue("Name"), queueCounterPointDetail.AttValue("QLenMax(Current,
"+str(timeInt) + ")")]
            queueLengthOutput.append(queueLengthData)

        # Volume measurements at different points
```

122

```python
        dataCollectionMeasurementPoints =
Net.DataCollectionMeasurements.GetMultiAttValues("Name")
        '''
        1 = S-P IN/OUT
        4 = M-S OUT,M-S OUT
        5 = P-S OUT
        10 = M-P OUT,M-P OUT
        11 = P-M OUT,P-M OUT
        12 = S-M OUT,S-M OUT
        '''
        outputDataCollectionMeasurementPoints = [t for t in dataCollectionMeasurementPoints if t[0] in
[1,4,5,10,11,12]]
        for outputDataCollectionMeasurementPoint in outputDataCollectionMeasurementPoints:
            dataCollectionMeasurement =
Net.DataCollectionMeasurements.ItemByKey(outputDataCollectionMeasurementPoint[0])
            attribute = 'Vehs(Current,'+ str(timeInt) + ', All)'
            data = [i, timeInt, outputDataCollectionMeasurementPoint[0],
dataCollectionMeasurement.AttValue("Name"), dataCollectionMeasurement.AttValue(attribute)]
            output.append(data)

    columns = ['item','TimeInt', 'data collection measurement index', 'Data Collection Measurement',
'Simulated Volume']
    outputDataFrame = pd.DataFrame(output, columns=columns)
    newMergedDataFrame = pd.merge(outputDataFrame, Data,  how='inner', left_on=['TimeInt','Data
Collection Measurement'], right_on = ['TimeInt','Data Collection Measurement'])
    newMergedDataFrame["Error Percentage"] = (abs(newMergedDataFrame["Simulated Volume"] -
newMergedDataFrame["Input Data Vehs (All) (veh/hr)"])/newMergedDataFrame["Input Data Vehs
(All) (veh/hr)"]) * 100

    averageErrorCalculationForEachDataCollectionMeasurementPoints =
newMergedDataFrame.groupby(["data collection measurement index", "Data Collection
Measurement"]).agg({"Error Percentage": 'mean'}).reset_index()
    replacement_dict = {
        "S-P IN/OUT": 'S-P IN/OUT and M-P OUT,M-P OUT',
        "M-S OUT,M-S OUT": 'M-S OUT,M-S OUT and P-S OUT',
        "P-S OUT": 'M-S OUT,M-S OUT and P-S OUT',
        "M-P OUT,M-P OUT": 'S-P IN/OUT and M-P OUT,M-P OUT',
        "P-M OUT,P-M OUT": "P-M OUT,P-M OUT and S-M OUT,S-M OUT",
        "S-M OUT,S-M OUT": "P-M OUT,P-M OUT and S-M OUT,S-M OUT"
        }
    averageErrorCalculationForEachDataCollectionMeasurementPoints["CombinedMeasurementPoint"]
= averageErrorCalculationForEachDataCollectionMeasurementPoints["Data Collection
Measurement"].replace(replacement_dict)
    combinedOutputData =
averageErrorCalculationForEachDataCollectionMeasurementPoints[["CombinedMeasurementPoint",
"Error Percentage"]].groupby("CombinedMeasurementPoint").agg({"Error Percentage":
"mean"}).reset_index()

    # Queue Counter Output Details:
    queueCounterClumns = ['TimeInt', 'Queue Counter index', 'Queue Counter Name', 'QLenMax']
    queueCounterOutputDataFrame = pd.DataFrame(queueLengthOutput,
columns=queueCounterClumns)
    queueCounterOutputMaxValues = queueCounterOutputDataFrame.groupby('Queue Counter
index')['QLenMax'].max()
    queueCounttOutputResult = queueCounterOutputDataFrame.groupby(['Queue Counter index',
'Queue Counter Name']).agg(maximumQueueLength=('QLenMax',  'max')).reset_index()

    ACTUAL_MAXIMUM_QUEUE_LENGTH = [230.53, 143.93, 198.13] #This data is measured
values and is  used for error calculation on queue length
```

```python
    queueCounttOutputResult['Error Percentage'] =
abs((queueCounttOutputResult['maximumQueueLength'] -
ACTUAL_MAXIMUM_QUEUE_LENGTH)/ACTUAL_MAXIMUM_QUEUE_LENGTH) * 100


    # print output:
    filename =
f"D:\Output\FINAL_output_ga_005_sin_2_percent_error_20_saturation\generation_{ga_instance.gene
rations_completed}.xlsx"
    sheet_name = f"solution_{solution_idx}_output"
    if os.path.isfile(filename):
        with pd.ExcelWriter(filename, engine="openpyxl", mode="a", if_sheet_exists= "overlay") as
Writer:
            outputDataFrame.to_excel(Writer, sheet_name= sheet_name, index= True)
            newMergedDataFrame.to_excel(Writer, sheet_name= sheet_name, startrow= 30, index =
True )
            averageErrorCalculationForEachDataCollectionMeasurementPoints.to_excel(Writer,
sheet_name= sheet_name, startrow= 60, index = True)
            combinedOutputData.to_excel(Writer, sheet_name = sheet_name, startrow= 70, index =
True)
            queueCounterOutputDataFrame.to_excel(Writer, sheet_name= sheet_name, startrow= 80,
index= True)
            queueCounttOutputResult.to_excel(Writer, sheet_name= sheet_name, startrow= 100, index=
True)
    else:
        with pd.ExcelWriter(filename, engine="openpyxl", mode="w") as Writer:
            population = ga_instance.population
            populationDf = pd.DataFrame(population, columns= ['LookBackDistMax',
'LookAheadDistMin', 'LookBackDistMin', 'W74ax', 'W74bxAdd', 'W74bxMult', 'MinFrontRearClear',
'LatDistStandDef', 'LatDistDrivDef'])
            populationDf.to_excel(Writer, sheet_name= "parameter", index = True)
            outputDataFrame.to_excel(Writer, sheet_name= sheet_name, index= True)
            newMergedDataFrame.to_excel(Writer, sheet_name= sheet_name, startrow= 30, index = True )
            averageErrorCalculationForEachDataCollectionMeasurementPoints.to_excel(Writer,
sheet_name= sheet_name, startrow= 60, index = True)
            combinedOutputData.to_excel(Writer, sheet_name = sheet_name, startrow= 70, index = True)
            queueCounterOutputDataFrame.to_excel(Writer, sheet_name= sheet_name, startrow= 80,
index= True)
            queueCounttOutputResult.to_excel(Writer, sheet_name= sheet_name, startrow= 100, index=
True)
    averageError = combinedOutputData.agg({"Error Percentage": "mean"}).values[0]
    averageQueueLengthError = queueCounttOutputResult.agg({"Error Percentage": "mean"}).values[0]

    # Fitness value
    fitness = 1/averageError
    queueLengthFitness = 1/averageQueueLengthError
    return fitness


if __name__ == "__main__":
#=============================================================================
=======
    # Candidate parameter sets
    LookBackDistMax = [80, 120, 150, 180]
    LookAheadDistMin = [10, 15, 20, 25, 30]
    LookBackDistMin = [5, 10, 15, 20, 25, 30]
    W74ax = [0.3, 0.7, 1, 1.5, 2]
    W74bxAdd = [0.1, 0.5, 1, 1.5, 2]
    W74bxMult = [0, 1, 2, 3]
    MinFrontRearClear = [0.1, 0.3, 0.5, 0.7, 1]
```

```python
    LatDistStandDef = [0.1, 0.2, 0.3, 0.4, 0.5]
    LatDistDrivDef = [0.6, 0.7, 0.8, 0.9, 1]


    Pairs = [LookBackDistMax, LookAheadDistMin, LookBackDistMin, W74ax, W74bxAdd,
W74bxMult, MinFrontRearClear, LatDistStandDef, LatDistDrivDef]
    PairsLength = [len(LookBackDistMax), len(LookAheadDistMin), len(LookBackDistMin),
len(W74ax), len(W74bxAdd), len(W74bxMult), len(MinFrontRearClear), len(LatDistStandDef),
len(LatDistDrivDef)]
#=====================================================================
=======
# Latin Hypercube Sampling Design
# Candidate sets
    sam = 20 # sample size
    LHsets = lhs(9, samples = sam)                          # Generate candidate sets
    LHsets = norm(loc=0, scale=1).ppf(LHsets)       # Normalize the value of candidate sets to N(0,1)
    for i in range(9):                         # Substitute the LH matrix to real value
        Range = Pairs[i]
        N = PairsLength[i]
        Prob = 1/N
        Interval = [];
        for k in range(N-1):
            Interval.append(norm.ppf(Prob*(k+1)))
        for j in range(sam):
            for q in range(N-1):
                if Interval[q] > LHsets[j,i]:
                    LHsets[j,i] = Range[q]
                    break
                if q == N-2:
                    LHsets[j,i] = Range[N-1]


    columns = ['LookBackDistMax', 'LookAheadDistMin', 'LookBackDistMin', 'W74ax', 'W74bxAdd',
'W74bxMult', 'MinFrontRearClear', 'LatDistStandDef', 'LatDistDrivDef'];

#This part stores the initial parameters into csv file.
    parameterData = pd.DataFrame(LHsets, columns= columns);
    parameterData.to_csv("D:\Output\geneticAlgorithmParameterData.csv", header= columns);
#=====================================================================
=======

# This portion performs reading field input volumes for error calculation.
    Data = pd.read_excel('D:\Output\StandardMeasuredOutputUpdatedVersion.xlsx', sheet_name=
'GEH Check', header= 0)
    conditions = [(Data['Time Int'] == "0-900"), (Data['Time Int'] == "900-1800"), (Data['Time Int'] ==
"1800-2700"), (Data['Time Int'] == "2700-3600")]
    values = [1, 2, 3, 4]
    Data['TimeInt'] = np.select(conditions, values)

# VISSIM Configurations
    # Load VISSIM Network
    Vissim = com.Dispatch("Vissim.Vissim");
    Vissim.LoadNet(r'D:\Engineering\Master of Science in Transportation Engineering\4th Sem -
Thesis\03. Thesis\02. VISSIM Analysis\03. Singhadurbar - Latest Final GA - Error 2%\Singhadurbar -
Latest Final 2_percent_error.inpx');
    # Define Simulation Configurations
    Sim = Vissim.Simulation
    Net = Vissim.Net
    G = Vissim.Graphics
    dbpss = Net.DrivingBehaviors;                          # Driving behavior module
    dbps = dbpss.ItemByKey(1);                              # Urban (motorized)
```

```
    # Set Simulation Parameters
    TotalPeriod = 3900;        # Define total simulation period
    WarmPeriod = 300;          # Define warm period 5 minutes
    step_time = 10             # Define Step Time
    total_number_of_stem_in_each_simulation = TotalPeriod * step_time
    Sim.SetAttValue('SimPeriod',TotalPeriod)
    Sim.SetAttValue('SimRes',step_time)
    Sim.SetAttValue('NumRuns',1)
    G.CurrentNetworkWindow.SetAttValue('QuickMode',1)


#=============================================================================
=======
# Range for simulation parameters
    LookBackDistMaxRange = [80, 180]
    LookAheadDistMinRange = [10, 30]
    LookBackDistMinRange = [5, 30]
    W74axRange = [0.3, 2]
    W74bxAddRange = [0.1, 2]
    W74bxMultRange = [0, 3]
    MinFrontRearClearRange = [0.1, 1]
    LatDistStandDefRange = [0.1, 0.5]
    LatDistDrivDefRange = [0.6, 1]

    inputParameterRange = np.stack([
      LookBackDistMaxRange,
      LookAheadDistMinRange,
      LookBackDistMinRange,
      W74axRange,
      W74bxAddRange,
      W74bxMultRange,
      MinFrontRearClearRange,
      LatDistStandDefRange,
      LatDistDrivDefRange
      ], axis=0)

    init_range_low = inputParameterRange[:,0]
    init_range_high = inputParameterRange[:, 1]
    gene_space = [
      {"low": LookBackDistMaxRange[0], "high": LookBackDistMaxRange[1]},
      {"low": LookAheadDistMinRange[0], "high": LookAheadDistMinRange[1]},
      {"low": LookBackDistMinRange[0], "high": LookBackDistMinRange[1]},
      {"low": W74axRange[0], "high": W74axRange[1]},
      {"low": W74bxAddRange[0], "high": W74bxAddRange[1]},
      {"low": W74bxMultRange[0], "high": W74bxMultRange[1]},
      {"low": MinFrontRearClearRange[0], "high": MinFrontRearClearRange[1]},
      {"low": LatDistStandDefRange[0], "high": LatDistStandDefRange[1]},
      {"low": LatDistDrivDefRange[0], "high": LatDistDrivDefRange[1]}
    ]


# This method will be called at the initiation of GA.
    def on_start(ga_instance):
      print("Starting generation {generation}".format(generation=ga_instance.generations_completed))


# This method will be called after the completion of each fitness calculation of all chromosomes of
population set.
    def on_generation(ga_instance):
      # Get the current generation number
      generation = ga_instance.generations_completed
```

```python
        # Get the fitness values of the current population
        fitness_values = ga_instance.last_generation_fitness

        fitness_values_df = pd.DataFrame(fitness_values, columns=["Fitness"])
        fitness_values_df["Average Error"] = 1/fitness_values_df["Fitness"]

        # Storing population generated by GA for each generation.
        filename =
f"D:\Output\FINAL_output_ga_005_sin_2_percent_error_20_saturation\generation_{ga_instance.gene
rations_completed}.xlsx"
        sheet_name = "fitness_output"
        if os.path.isfile(filename):
            with pd.ExcelWriter(filename, engine="openpyxl", mode="a", if_sheet_exists= "overlay") as
Writer:
                fitness_values_df.to_excel(Writer, sheet_name=sheet_name, index = True)
        else:
            with pd.ExcelWriter(filename, engine="openpyxl", mode="w") as Writer:
                fitness_values_df.to_excel(Writer, sheet_name=sheet_name, index = True)


        # Print generation number completed and fitness values of all chromosomes of population used in
each generation
        print("Generation:", generation)
        print("Fitness Values:", fitness_values)

# Process the data, preparing inputs for GA
    geneticAlgorithm = pygad.GA(
        num_generations=20,
        num_parents_mating = 10,
        fitness_func = fitnessFunction,
        initial_population = LHsets,
        num_genes= 9,
        init_range_low= init_range_low,
        init_range_high= init_range_high,
        gene_type = [float, 2],
        parent_selection_type= "rws",
        crossover_type= "two_points",
        crossover_probability= 0.8,
        mutation_type= "random",
        mutation_probability=0.05,
        mutation_by_replacement=False,
        mutation_percent_genes='20',
        mutation_num_genes=2,
        random_mutation_min_val=-1.0,
        random_mutation_max_val=1.0,
        gene_space=gene_space,
        allow_duplicate_genes=False,
        on_start=on_start,
        on_fitness=None,
        on_parents=None,
        on_crossover=None,
        on_mutation=None,
        on_generation=on_generation,
        on_stop=None,
        delay_after_gen=0.0,
        save_best_solutions=False,
        save_solutions=False,
        suppress_warnings=False,
        stop_criteria=["reach_0.5", "saturate_20"],
```

```
    parallel_processing=None,
    random_seed=None,
    logger=None,
    )

geneticAlgorithm.run()

best_solution = geneticAlgorithm.best_solution()
print(best_solution)
```

APPENDIX H Pictures of Field Survey