CE 495 B.Tech. Project Stage-II Report on

Developing Simulation model for Indian Road network using TRANSIMS

Submitted by

Venktesh Pandey

(Roll No. 100040090)

Under the supervision of

Prof. Gopal.R.Patil

Department of Civil Engineering

Indian Institute of Technology Bombay

3 April, 2014

Acknowledgement

I would like express my sincere thanks and gratitude to **Prof. Gopal R. Patil** and **our research group** for their continuous and unfailing support, guidance and help, which have been invaluable during the course of this project. Their knowledge, insight and constant motivation at each step of the project has been instrumental in its proceedings.

Special thanks to Neeraj Saxena, Harshala sardar, Ritesh Sharma, Renal Chheda and Prateek Bansal for their crucial help at moments and support. I would also like to thank my friends and parents for providing kind support during different times.

Lastly all credit to almighty for keeping me move even while deepest of lows and highest of peaks.

CONTENTS

List of Figures

List of Tables

Chapter 1- Introduction

1.1 General

Transportation Analysis and Simulation System or TRANSIMS is an integrated set of tools developed to conduct regional transportation system analysis which was initially developed at Los Alamos National Laboratory.TRANSIMS as a software follows agent based cellular automata model used for creating activity based travel demand forecasting models. TRANSIMS generates the activity patterns for the population of a study area and even develops the routing decisions taken by an individual. The activity based modeling which has been incorporated in TRANSIMS is often sought over the traditional Four Stage Modeling (FSM) by the several Metropolitan Planning Organizations (MPO).

TRANSIMS has found numerous applications in the US on implementing several policy measures like congestion pricing, Intelligent Transportation System, etc. It has been successfully tested for doing the regional planning of district of Alexandria. TRANSIMS has also found its application in chalking out a multimodal emergency evacuation plan for the New Orleans Louisiana metropolitan region. Most of the MPO's these days maintain the data required for carrying out simulation using TRANSIMS and check the feasibility of various measures.

TRANSIMS is capable of generating the simulation model from different data sources. The simulation process of TRANSIMS can be broadly subdivided into two heads, namely, Track-1 and Track-2 implementations. The Track-1 implementation takes in the OD matrix as an input, which is readily available with the planning organizations of the city. Whereas, Track-2 implementation takes in the socio-economic and individual activity pattern data as an input while constructing the activity patterns for the entire population.

TRANSIMS offers us following advantages -

a) Can simulate larger areas of network covering regional level and areas as large involving >100,000 links and more than 25,000,000 travellers

b) Can implement 24 hour simulation of a larger network under reasonable speed, and can iterate for convergence to equilibrium in shorter iterations
c) Writes desired output on network and vehicular characteristics to required level of details and can capture movement of individual vehicles at every point of time
d) Is open source and thus can be analyzed and modified to suit customized traffic conditions by making changes in the source-code
e) Implements efficiently both of Trip and activity based models(Track-1 and Track-2 implementation) and hence can even be applied to a city where previous travel demand modelling studies have not been conducted

Under the shade of given advantages, it is highly efficient to build a better simulation model for Indian road networks in TRANSIMS. That can be well used to analyze influence of different level of changes like congestion pricing, intelligent transport systems, emission control methods etc. TRANSIMS also is capable of building automatic signaling systems for different nodes based on capacity levels of the links, and thus can save us the task of not giving the input on different level of signal details at intersections.

However to apply TRANSIMS for Indian road networks pose the following problem –

a) Heterogeneous traffic conditions with varying types of vehicles
b) Left-side driving
c) Unorganized movement of traffic and no strict lane following implementation

To get into modifying TRANSIMS to get above steps working we need a conscious approach for analysis and understanding of each and every module and understand how the changes in the existing modules can bring about the change we expect to see.

1.2 Objectives of Study

Explaining the purview of TRANSIMS before, following were the objectives of the study –

- Understanding the working of different modules involved in the TRANSIMS Track-1 implementation
  a) Special focus given to analyze how different parameters cause the change we want to implement
  b) Understanding how parameters can be adjusted to meet the desired level of details in the simulation output

- Identifying the elements where changes need to incorporated for simulating Left-side driving conditions in TRANSIMS
    a) Testing for changes in parameters that influence the calculation of travel time and travel cost used in next iteration for routing
    b) Comparing the TRANSIMS network results to analyze the same for left side driving and flipped network condition
- Building a desired visualization of the network simulation in TransimsVIS (developed by simulation using TRANSIMS Studio platform)
- Improving the existing Track-1 implementation model of IIT-Bombay campus with left side driving characteristics and calibrated parameters
- Concluding about how the changes be formalized and extended for large scale networks

The major part of work was to understand and continue the previous work done by Saxena (2013) on developing new vehicle type and Track-1 & Track-2 simulation model of IIT-Bombay campus using TRANSIMS.

1.3 Organization of Report

The entire report has been divided into 10 chapters –

Chapter-1 deals with introducing the problem which is to be addressed throughout the report. It talks about the objectives and how TRANSIMS can help in accomplishing those

Chapter-2 brings in the brief literature review of the relevant contents which helped me during the course of this project. Chapter-3 shifts the focus to the details of transims which were studied for extensive amount of time to develop a thorough understanding of the working of software.

Chapter-4 is a journey through the conceptual explanation of why left side driving differs from right side driving and how the required changes can be implemented in TRANSIMS. Chapter-5 explains the various experiments which were done for bringing in the required adaptation in TRANSIMS. It offers observation and conclusion for different studies conducted on network.

Chapter-6 talks about crucial elements focused for re-implementation of Transims track-1, on IIT-Bombay campus. It talks about detailed input given to ever utility to get what we desire from the simulation. Chapter-7 talks about flipped network analysis, its comparison with the observed field data and then the calibration of penalty parameters for suitable match with the left-side driving scenario.

Chapter-8 titled summary and conclusions bring about the summary of the work explained before. Chapter-9 deals with the work that needs to be accomplished in future under future scope. Chapter-10 lists out the relevant references used though the text.

Chapter 2 – Literature Review

2.1 General

TRANSIMS is mainly an activity based travel demand modeling software which is used for conducting regional transportation system analysis. Due to this nature of the software package, it helps in analysing the impacts of introducing several policy measures like congestion pricing, tolling, high-occupancy vehicle lanes, dynamic driver information systems, and other Intelligent Transportation System (ITS) devices that aim to ease the travel experience. Similar to the four-step transportation model, TRANSIMS is a sequential (not simultaneous) model which means that there is an explicit feedback loop whereby the individual travelers' reactions to information about the satisfaction of their preferences is modeled. The key difference between the 4-stage modelling approach and TRANSIMS approach is that TRANSIMS is microsimulation based and is therefore capable of modeling the stochastic and dynamic attributes of the transportation system. In TRANSIMS, all of the major lifestyle and travel decisions such as what activities to participate in, when to participate in the activities, where to participate, what mode to use, or what route to choose can be made in a consistent manner at the individual traveler level. This part of chapter analyses how TRANSIMS have been implemented in different parts of the world and how the different level of analysis be done using the software.

2.2 Simulation basics analysis

Lieberman and Rathi (2001) explain what makes developing a simulation model necessary and important for different level of analysis. They justify how traffic simulation models can 'satisfy a wide range of requirements' including – evaluation of alternative treatments, testing new designs, as an element of design process, for embedding in other tools, for training personals and for safety analysis. Need of simulation model justified in it tells that Transportation modelling indeed is best analyzed using simulation because of following reasons –

- Mathematical treatment of a problem is infeasible or inadequate due to its temporal or spatial scale, and/or It must be emphasized that traffic simulation, by itself, cannot be the complexity of the traffic flow process
- The assumptions underlying a mathematical formulation (e.g., a linear program) or an heuristic procedure (e.g., those in the Highway Capacity Manual) cast some doubt on the accuracy or applicability of the results
- The mathematical formulation represents the dynamic traffic/control environment as a simpler quasi steady- state system

Also is explained in the paper on how to differentiate microscopic – mesoscopic -macroscopic simulations. By the conclusions from that microscopic simulation takes into account individual vehicle position, we conclude TRANSIMS incorporates a microscopic simulation of the traffic.. A proper classification system developed in the paper help us conclude that TRANSIMS is a -

- Discrete time simulation model – as every motion is observed after a fixed time-step
- Stochastic simulation – involves random variations in speed and driver behavior
- Microscopic simulation model – covering up intricate details of every vehicle at every time-step (however the paper mentions it to be a mesoscopic model, other instances prove it wrong)

Fidelity of the simulation needs to be verified by sensitivity analysis of all the parameters, which will depend on the case study where we are applying TRANSIMS on.

2.3 Dynamic Traffic Assignment- analysis

The primer developed on *Dynamic Traffic Assignment* by Transportation Research Board(2011 June), gives clear insight on how a dynamic traffic assignment takes into account the congestion effect to decide suitable route for the vehicles during traffic assignment. It also clarifies that dynamic system equilibrium as the condition for convergence of assignment procedure, is achieved when in every time epoch travel cost for users on all used paths is equal.

TRANSIMS incorporates the dynamic traffic assignment and calls its routing procedure after each iteration which decides different routes of vehicles departing at different times, based on the congestion level obtained on those links at the times when the vehicle would be reaching the link entrance to travel on it (called experienced travel time behavior).

Fig.2.1 justified in the paper tells how the experienced travel time plays the role of deciding the shortest travel cost path. In part E(a) of the figure it can be clearly understood that vehicle departing at time1 from node 1, will reach node 2 only after 1 time step. Thus what it'll experience on link 2→ 4 is the travel time at time step-2, which is '2 sec'. Similar methodology when followed yield different results for travel costs between different OD pairs.

Also is evident form Fig.2.1 that though the instantaneous travel time criteria considers different travel time at different points of time, it doesn't take into account time at which vehicle reaches the entrance of link. Because of which the travel costs calculated for different routes is different in both the cases.

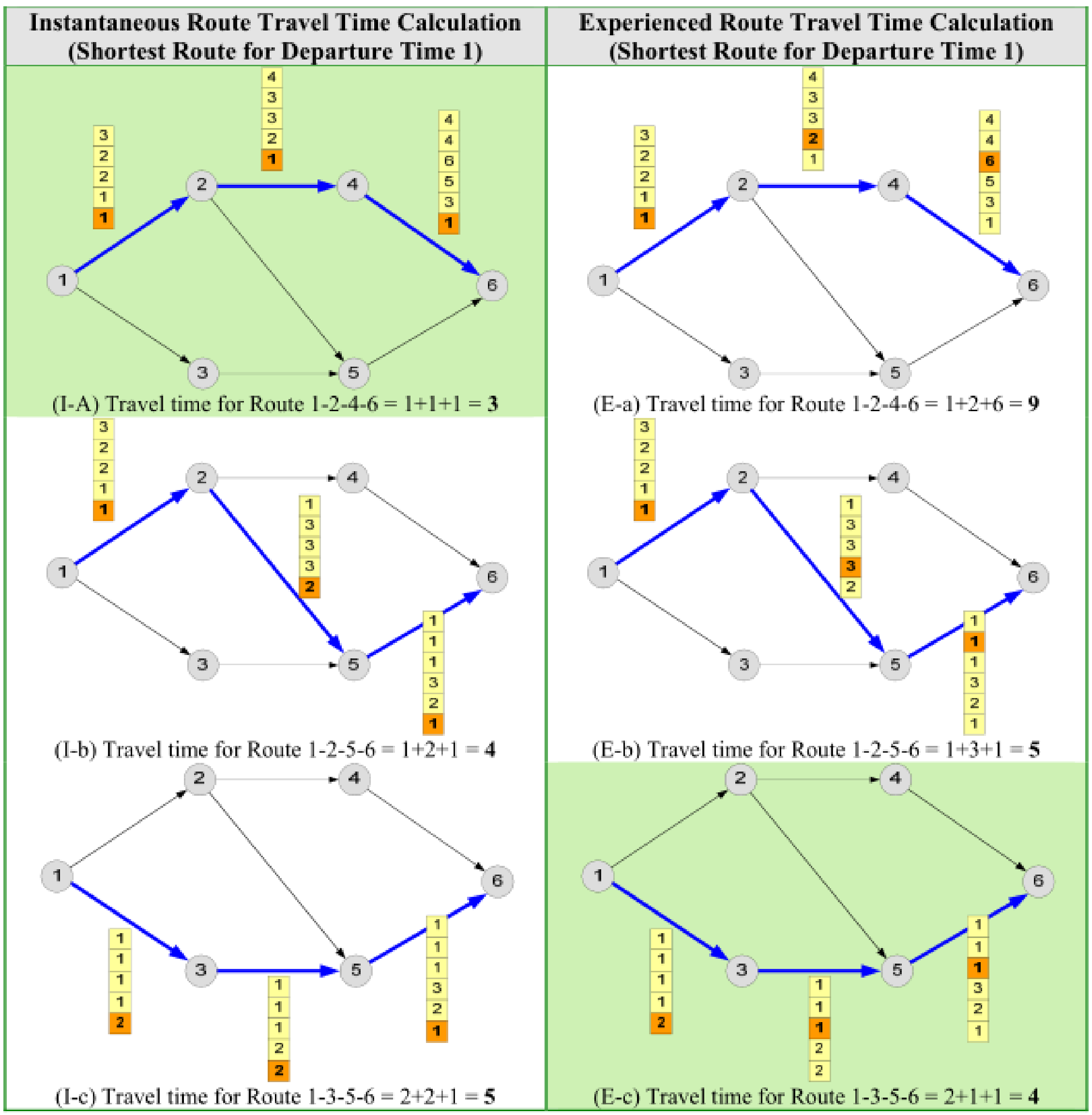


Fig. 2.2 Experienced vs Instantaneous travel time (Source [1])

TRANSIMS consider each time-step of 15min when deciding for travel time in routing for next iteration. Regression model is fit for every time step of 15min and is used to determine the travel time at any point in that interval.

2.4 Case studies under TRANSIMS

TRANSIMS being capable of simulating very large scale networks have been tested and calibrated for different cities across the continents. TRANSIMS has found numerous applications in the USA for conducting travel demand forecasting at a network level. Following case-studies lsit the applicability of TRANSIMS at different stages of work –

a) Wolshon and McArdle (2009) developed a multimodal emergency transportation model using TRANSIMS for the New Orleans Louisiana metropolitan region after the failure in evacuating the population during the hurricane Katrina in 2005. The study found out that TRANSIMS accurately predicted the spatial and temporal distribution of traffic during the evacuation.

b) Lawe et al. (2009) developed a track-1 implementation model followed by the calibration of TRANSIMS model in Chittenden County, Vermont, USA. The study area has a population of 145,000 with an urban sprawl of 1606 sq. km. The study found out that the track-1 model provided results with reasonable accuracy. The sensitivity analysis of the model was also conducted and the developed model was found to be stable. TRANSIMS has found many applications for developing a demand model for the small MPOs having a population up to 100,000.

c) Jeihani & Ardeshiri (2011) developed two models namely, the enhanced FSM or the Sub-TAZ model and the TRANSIMS Track-1 model. The study area chosen for modeling was a small network in Fort Meade, Maryland, USA comprising of 28 TAZ and 327 links. The study found out that the application of TRANSIMS model was better than the FSM because of its Dynamic Traffic Assignment capability. A track-2 model was also developed for the study area but was found to under-estimate the external zone trips.

d) Ullah et al. (2011) carried out a study to determine whether TRANSIMS would be able to provide the necessary tools to achieve the functional requirements of travel demand modeling for a several small sized MPO in Illinois, USA. The TRANSIMS Track-1 model was developed and the results were compared with the existing FSM. It was observed that the TRANSIMS model predicted values which were close to the field data.

e) As per the personal communication (Berryman M. September 10, 2013), TRANSIMS applicability was already been worked out at in Sydney, where left side driving is in practice. It explained that they worked by flipping the network by changing the x-coordinates to its negative and then flipping them back while analyzing the results. The work on Sydney simulated only the Router part of TRANSIMS for getting a coarse estimate on travel time parametric values.

f) The paper by Shan Huang & et.al (2012), describes precisely the crucial difference to be observed while modelling a university campus as compared to modelling a large scale region. Following key-aspects were found to be important –

   i) Differences in trip making characteristics as compared to single family household trips

   ii) Significant proportion of trips completed by a particular service

Chapter 3- Details of TRANSIMS

3.1 General

Different modules of TRANSIMS have been well covered in documentation provided by the software professionals. Extensive details about each of those could also be found on Saxena (2013) *pg. 8-23* and its Annexure – D also covers up the basics of installation and working on each of the utility. A basic flowchart of different modules in TRANSIMS can be presented in Fig.3.1 –

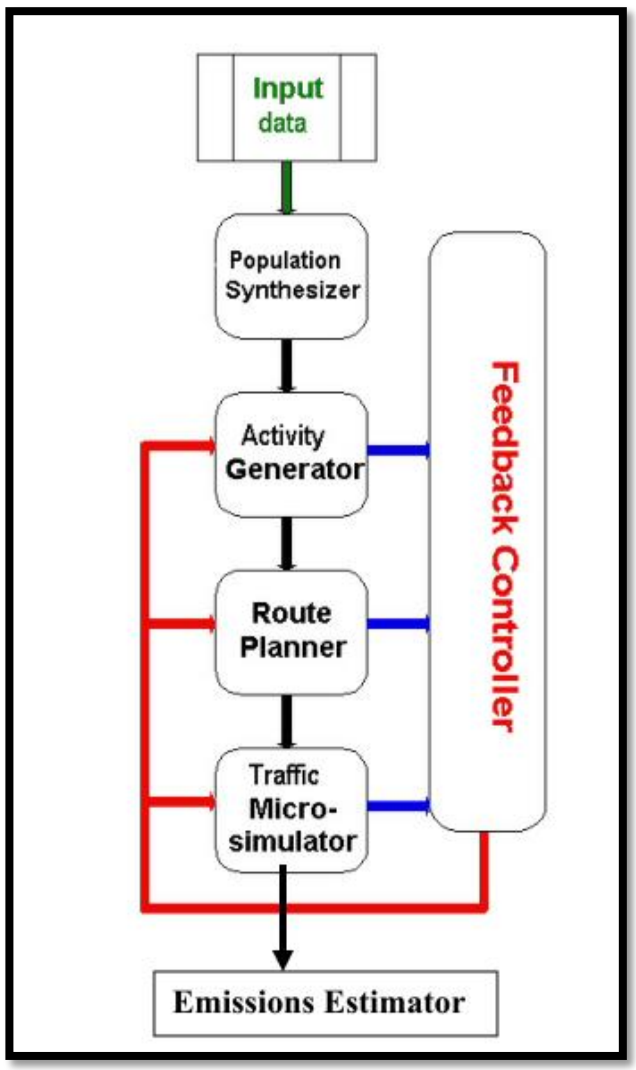


Fig.3.1 TRANSIMS Overview

The brief property of TRANSIMS when compared with other softwares reveals, it –

a) Does fine resolution modelling – discrete level
b) Large scale (can model everything from highways to local streets)
c) Low fidelity – less detailing wrt movement of vehicle and less complex

Even the velocities in TRANSIMS modelling are discrete to the quantum of cells/second, i.e. velocity of vehicle can only be 7.5mt/sec or 15m/sec etc. Following Figure.3.2 shows the correlation of TRANSIMS with a 4-stage model –

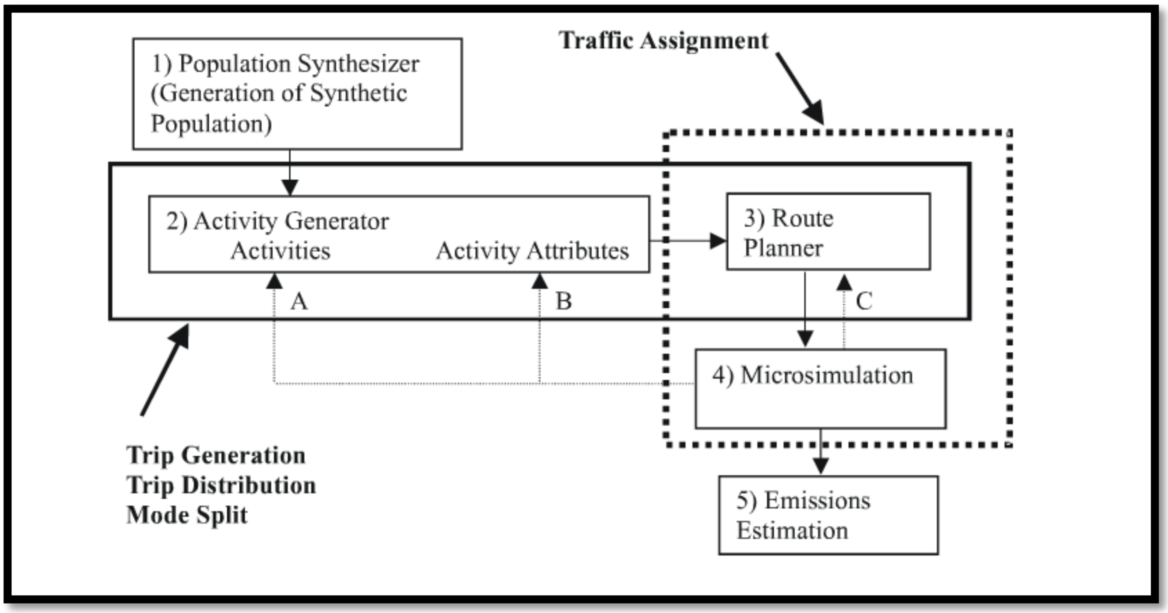


Figure 3.2 Comparison with 4-stage modelling

However for the study conducted a detailed analysis of how micro-simulator models and routing algorithm needs to be done, which have been presented in the further sections.

3.2 Micro-simulator Algorithm

Micro-simulator is one of the most important utility of TRANSIMS which brings into effect the real picture of simulation of vehicles on road. It is one of the utility that's calculates the network performance under dynamic influences of vehicular movements and changing network parameters. Basic steps in micro-simulator comprise of following details –

a) Reading the input files on network and vehicular locations
b) Converting the network into cell grids and then summarize link connections and initialize traffic controls

c) Construct a travel plan for each transit route and then synthesize a vehicle for each run

d) Loop in through each time step and then –

- When time step equals a whole second, process output increments for the second, update new network restrictions, update signal timings plan and initiate transit runs scheduled to start during the second
- Process vehicles in the priority queue
- Process different lane change requests
- Process the vehicle loading queue
- Again when time step equals whole second, read different travel legs from plan file that start at next second, convert travel leg to a leg-link travel plan and then load vehicle to the network or the travel queue
- Calculate movements for each traveler and each transit vehicle
- Move each vehicle to its new location and travel speed

e) Close the output files and print processing and problem summaries

This summary of algorithm gives us the idea of the precise movement of vehicle on road, which help us conclude the method to adopt for making changes for left-side driving.

3.3 DTA in TRANSIMS

Dynamic traffic assignment essentially works out the same purpose of bringing users to dynamic equilibrium by iterative feedback process between model components. In each iteration TRANSIMS adjusts activity locations, activity patterns, travel modes, travel schedules & or travel path of selected no. of individuals.

TRANSIMS modelling process uses DTA by changing two attributes – microsimulated vehicular movements and time-dependent network attributes. It basically has three effective phases at its back-end that perform the process of DTA step-wise. These include Router stabilization, Microsimulator stabilization and User equilibrium convergence.

Router stabilization generates minimum impedance paths through the network for individual travelers based on traditional volume delay function. It iterates till V/C ratio comes under 1.2, and that the network is assumed to be distributed according to capacities which forms the basis for equilibrium.

Micro-simulator stabilization phase minimizes the impact of bottleneck location and excessive congestion experienced by travelers by re-routing trips that pass through congested networks. It does network refinement to improve operations like lane connectivity, pocket lanes, transit scheduler etc., and also adjusts the signal timings based on intersection demand.

User equilibrium convergence phase minimizes the travel time or generalized cost of travelers by comparing paths used in simulation to minimum impedance path based on network performance, effectively bringing closer to equilibrium. Fig.3.2 establishes the generic procedure for DTA in TRANSIMS.

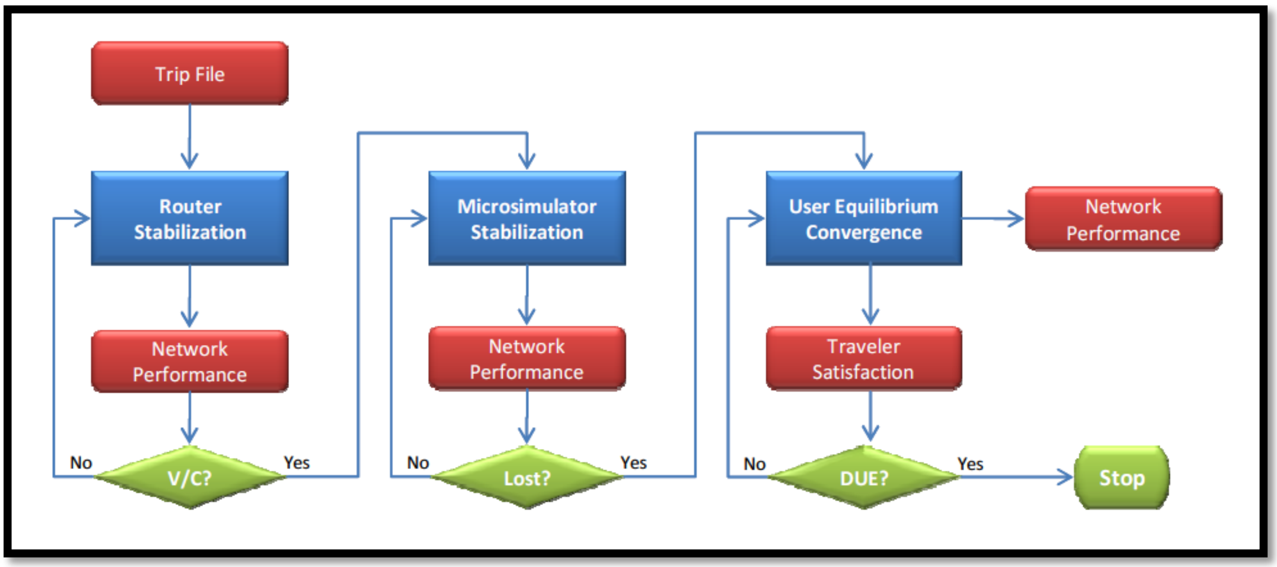


Fig. 3.3 DTA procedure in TRANSIMS

Because TRANSIMS simulates individual vehicles, the user equilibrium approach needs to be implemented in a different way. This is partially due to fact that Microsimulator is often unable to simulate the types of over capacity conditions generated by all-or-nothing assignments. When the demand exceeds the Micro-simulator's ability to process vehicles, queues form that can quickly cascade onto other links and result in system gridlock. The travel times generated by the Micro-simulator under these conditions tend not to be useful for making decisions about which travelers to re-route. They simply move the problem to another location on the network.

Thus TRANSIMS select some travelers and assign them a new plan, while keeping other travelers on the original plan as before. To select those key travelers TRANSIMS compares the difference in travel time obtained for each of the traveler and the travelers with highest difference (i.e. with most possibility of improving upon the travel time function) gets selected for route-change(or other preferences) in next iteration.

3.4 Critical outputs from Micro-simulator and Router iterations

Every software comes along with a varied options to get as output the crucial outputs of any traffic simulation. TRANSIMS being microscopic simulation records with it the detailed values for every time step. The major output files written by micro-simulator includes –

Table 3.1 Microsimulator output files

File Type	**Description**
Summary File	Aggregates volumes and performance statistics (e.g., travel time) for each link direction and turning movement by time increment (e.g., 15 minutes). Turning movement volumes and delays are optional
Snapshot File	Lists the link direction, offset, lane, and speed of each vehicle at specified time points (e.g. Every 5 minutes)
Occupancy	Lists the link direction, offset, lane, and cumulative occupancy of each cell by time increment (e.g., 15 minutes). May list the cells occupied at the maximum load point during the time increment or the total occupancy of each cell during the time increment
Event File	Lists the scheduled and actual time and link direction and offset for each traveler and trip event (i.e., start time and end time).
System Event File	Lists the time, node, and phasing information for each phase or timing plan change event at a traffic signal
Traveler File	Lists the link direction, offset, lane, and speed for each selected traveler by time step (e.g.,second)
Speed Bin	Aggregates the number of vehicles of a specified vehicle type by speed bin traveling on link segments at specified time increments

These outputs are written in a specified tab delimited text files which can then be discerned later

3.5 Transims-VIS

Among all the modules addressed in documentation provided in Saxena-2013, the details on this visualizer by TRANSIMS has not been covered. Thus visualizer seeks in the output files written after the simulator runs of Microsimulator and builds a compressed snapshot file locating the position of each and every vehicle at every point of time using python script in TRANSIMS studio.

This compressed file can then be viewed in the visualizer and vehicle to vehicle movement can be traced. This section describes how to run a sample network in TRANSIMS-studio which is a user-friendly platform for running python script files. Fig.3.3 shows a typical screen shot of the studio platform.

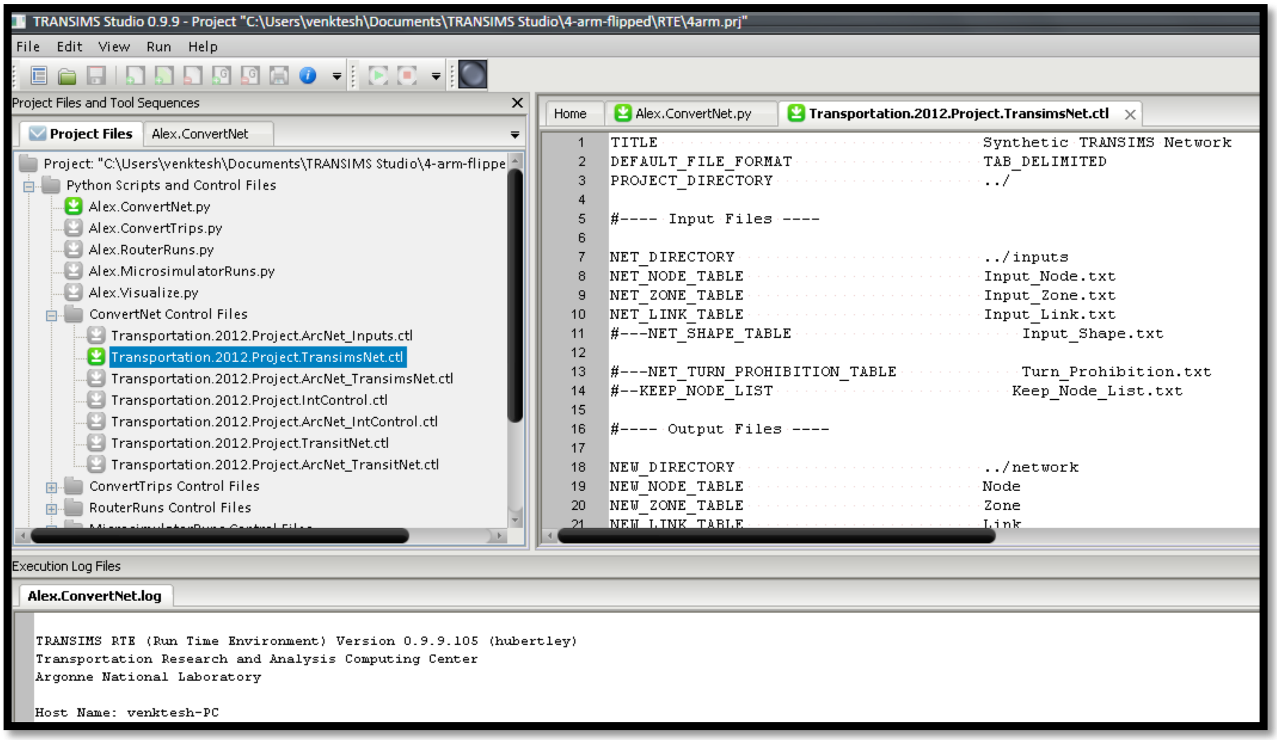


Fig.3.4 Transims Studio Platform

The platform shows control file in an organized format where they can directly be edited. The python script files just simply run the executables in a sequential format and are nothing but a replacement for batch file when in folder format.

To run the script files all the input files need to be arranged in suitable format. There are slight changes in many control files where some parameters are clichéd and not included. Then each utility is run turn by turn. There is an extra snapshotSeconds file written by Microsimulator output whose timings can be set directly in control file.

The output from each script are written in corresponding folder. At last an additional utility Visualize.py is run. This does the required conversion to compressed snapshot that TRANS-VIS uses to display the movement of vehicles. Later the control file reading that compressed snapshot is read by trans-VIS and it loads in the network and vehicles from the input given.

The following figures illustrate different possible visualizations offered by TRANS-VIS –

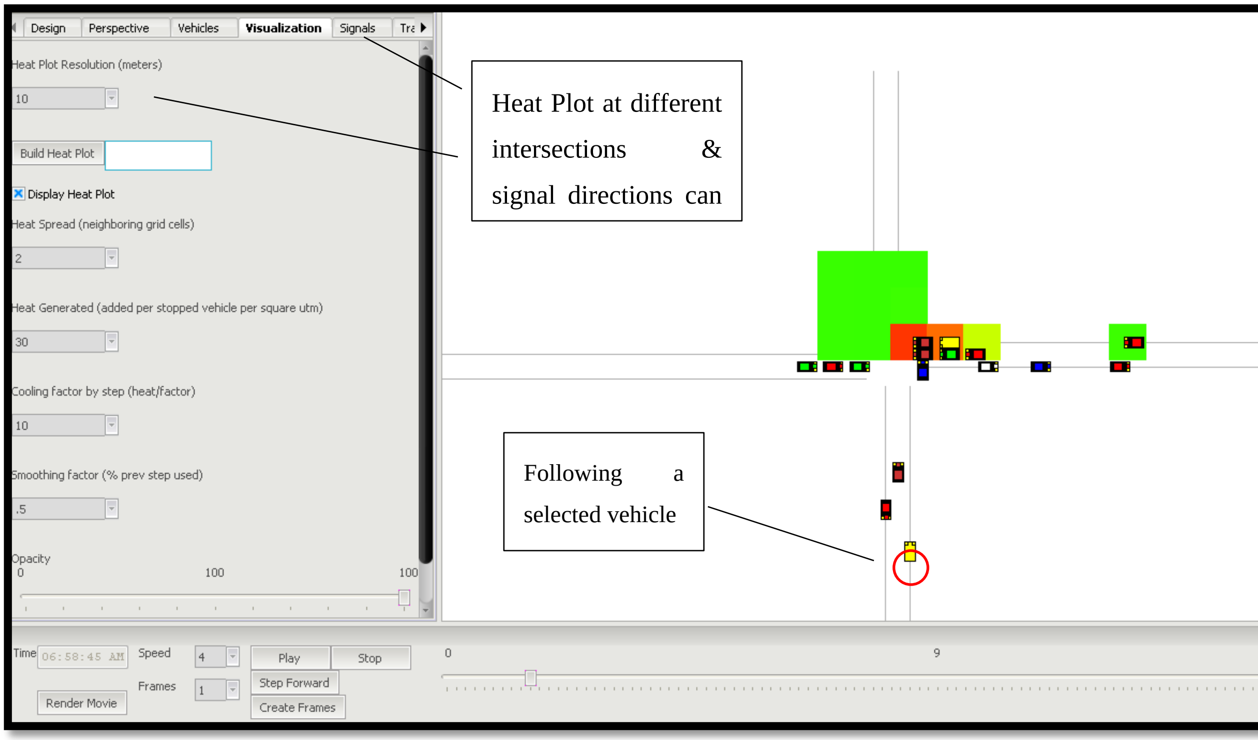


Fig. 3.5 Heat Plot in Transims Studio

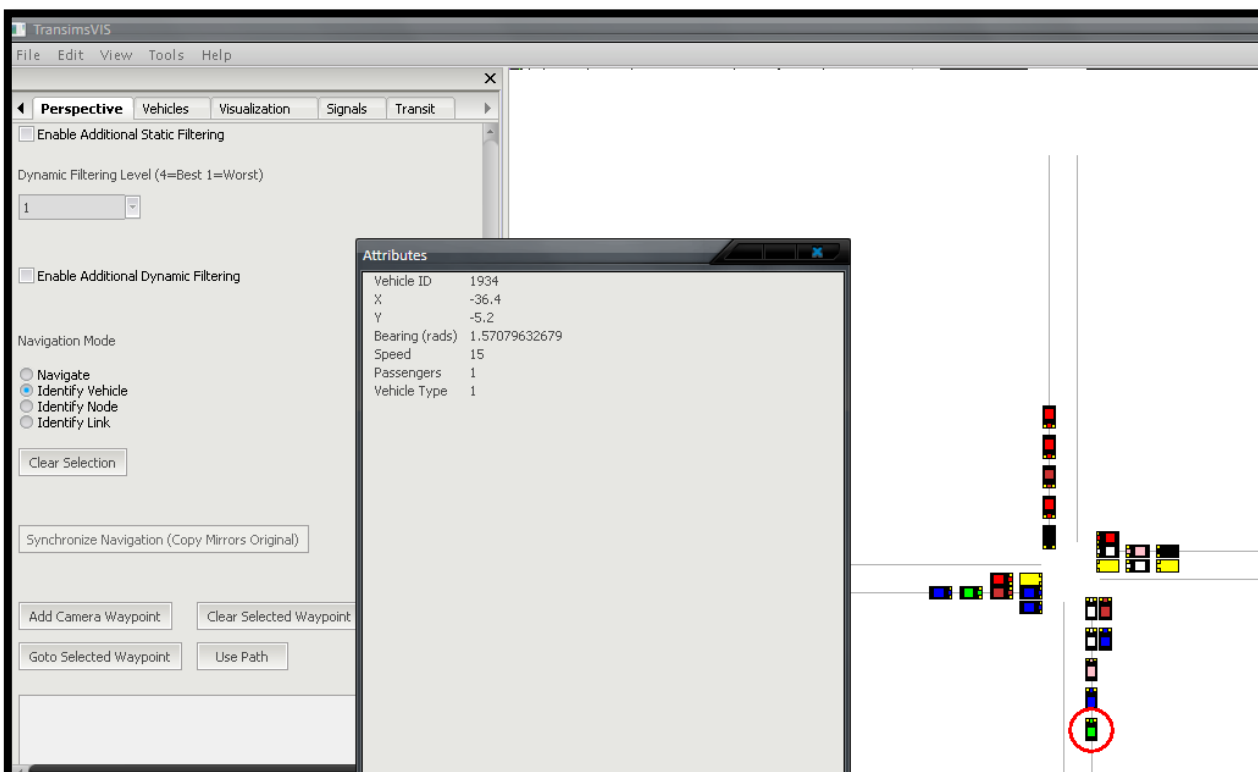


Fig.3.6 Following a vehicle in Visualizer

Though TRANS-VIS offers many utilities to analyze the network after the simulation run it is still difficult to tweak network and its parameters (node location, link length etc) before the simulation is run, for which analyzing the shape files written by arcnet.exe utility seems the only way.

Also don’t tend to get other statistics of the link (like current link-volume or total travel time for a vehicles or average travel time on the link etc) which we often are interested in and for which we still need to go back into the snapshots file to determine the .

However the advantages are still many in number since the visualization helps us verify the broader picture of congestion and vehicular volume, and give us a picture that the simulation indeed is running.

Chapter 4 – Explaining the Approach

4.1 General

In order to get TRANSIMS suited for Indian traffic conditions basic thought was given to the process which make driving in a country like India different from other countries of the world. India with its ever growing population has an altogether different traffic behavior and modelling those in a real time simulation system becomes a challenging task. Following points summarize the different areas which makes Indian traffic condition different from others-

- India is one of the country which enforces left-side driving as compared to 65%(1) of other countries in the world
- Highly heterogeneous behavior and highly chaotic movement. This comprises of highly varying static and dynamic characteristics of flow, difficulty in imposing lane use restriction and vehicles occupying any lateral position on the available road space

TRANSIMS as a software was designed to incorporate the right-side driving behavior as prevalent in USA. Though the applicability of the software makes it a highly popular software for its application on various level of detail in different countries, to make it work for conditions prevalent in India it needs to be modified and worked through at different level of details.

Under this approach it is identified what makes conditions in India different from other and what parameters to what level of details need to be modified for achieving closer realization of Indian traffic conditions.

4.2 Why Left vs. Right?

Though it's hardly a big issue which side the vehicle is driving on when considering movement of people, left-side driving case is indeed different from a right-side driving scenario. Following points cover up the details of the differences (references of TRANSIMS implementation have been used in the same) –

4.2.1 Travel distance at intersection –

It is indeed true that which side the vehicle is driving would make a difference in the travel time calculation for the vehicles movement through the intersection. If the path of the vehicle is fixed and at an intersection it has to take a left turn then a left-side driving scenario would lead the vehicle to choose for a shorter path than right-side driving scenario as illustrated in Fig.1

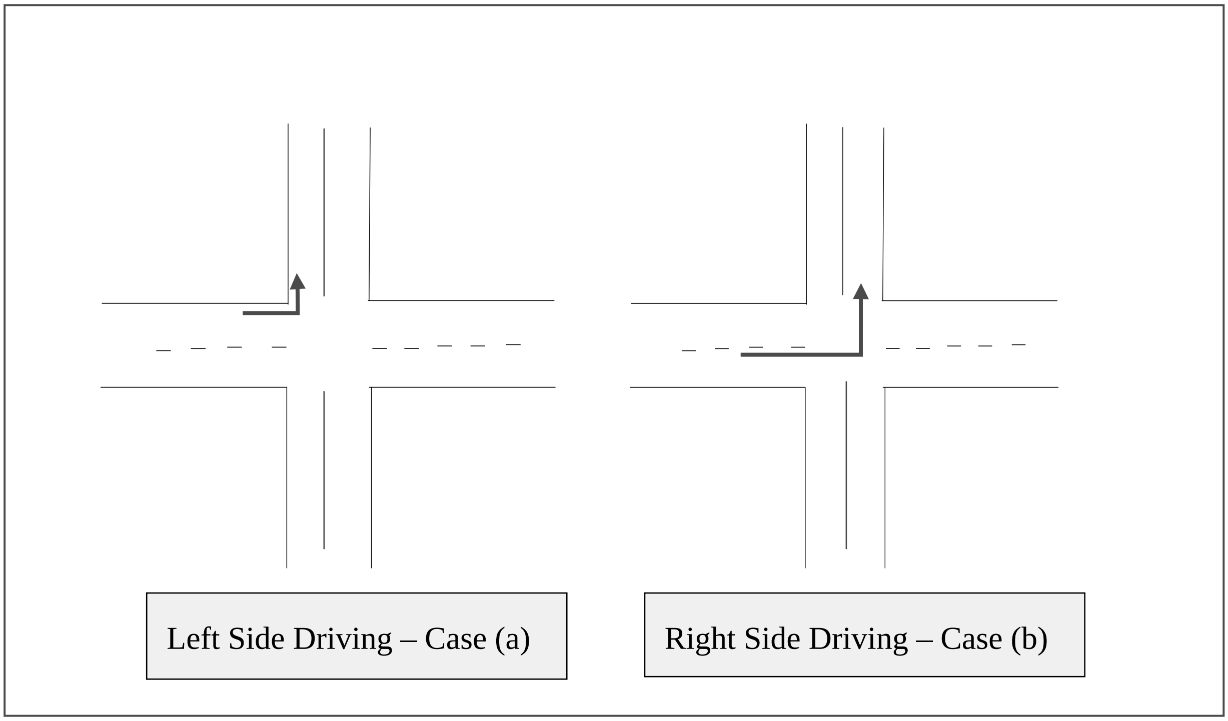


Fig.4.1 – Differences in driving pattern in left and right side driving scenarios

A vehicle travelling left in case (a) of Fig.1, will travel a shorter path and thus would take lesser travel time to traverse the intersections, as compared to the case (b). Thus travel times for vehicles would be influenced for sure.

Though the question that poses next is to what extent is that influence. The answer depends on the size of intersection and setback of the link from the node's center. Larger the intersection larger would be the differences as travel distance in case (b) would keep on increasing with larger intersections, however a negligible effect would be there for case (a).

4.2.2 Wait time at intersections

At many intersections in left-side driving we tend to have pocket lanes for movement of left-turning vehicles which tend to make their movement unrestricted whatever be the state of signal at the intersection, which is not the case in the right side driving where vehicle also needs to wait for extra time for signal in that direction to turn green for it to turn left. This addition to the travel time in terms of wait time at intersection could be enormously much (15-90sec per intersection) and thus would be a crucial parameter causing differences in travel time of vehicles in each scenario.

4.2.3 Movement on links

Even when the vehicle moves on the link travel behavior tends to be different for both the left and right turning scenario. This is described in terms of sticking onto the particular side of road and in terms of overtaking other vehicles.

A vehicle in a left side driving scenario tend to stick onto the left side of the road, while the reverse happens in the right side driving. This tend to make slower vehicles get concentrated on the left-most lanes in left side driving scenario.

Similar characteristics is observed when a vehicle tries to overtake. By traffic rules if movement in lanes is not restricted the vehicles in a left side driving scenario will tend to overtake the other vehicle only from the right side, which being opposite in right side driving would cause vehicles to reach a different approach lane at the intersection which lead to further small extra travel distance if lane-changing maneuver is not initiated when approaching the intersection.

4.3 Implementation in TRANSIMS

To approach towards the above mentioned differences in TRANSIMS, the major focus was put on the accounting for extra travel time addition due to movement at intersection. In effect if we want to convert right side driving algorithm to left side driving algorithm, we'll have to switch the way travel times are calculated for vehicles turning left and right in right side driving scenario, with travel time calculation for vehicles turning right and left in left side driving scenario respectively.

Why is this difference in travel time for two scenarios being stressed on? Clearly because of following two reasons –

- Travel time forms one of the major output of a simulation that can help us in comparison and validation of our model with respect to real data, so its accurate estimation is necessary
- In simulation with dynamic user equilibrium as the criteria for final route decision, travel cost of each of the route form an important input in deciding the weight of the route and amount of traffic to be assigned to that route. For accurate routing in subsequent iterations we need to accurately establish the travel time

How were different problems in left-side driving movement accounted for in TRANSIMS is explained further –

4.3.1 Switching the right with left

To ensure that travel distance at the intersection is corresponded the same way as would have been with left side driving, one needs to identify how inside TRANSIMS, can one switch the way travel times are calculated for different types of turning movement.

For this different experiments were conducted (listed in further sections) that attempted to yield conceptually the intended flipping of directions. The results and discussions for the same have been presented in the subsequent sections

This flipping would cover up the manner in which the travel distance and waiting times differ for same turning direction vehicles in both the scenario.

4.3.2 Overtaking and Lane-change

As mentioned before though the travel pattern on links differ on a non-restrictive lane following rules traffic, we are not posed by such a problem in TRANSIMS. This is because TRANSIMS follow organized lane behavior and believes that overtaking and lane change happen randomly based only on the speed of the vehicles on the road and the availability of the lanes for lane change. Whereas at intersections lane change is done as expected, with a left turning vehicle moving to the left of the lane at the approach link. Hence such a behavior is not much of a problem

Chapter 5 – Experiments for Adaptation in TRANSIMS

5.1 General

This chapter comprises various methodologies followed to get the required sorting approach as described in the previous chapter. All the tests were conducted on small test network which could be later tried and expanded for larger road networks. Following steps were followed to get the approximate simulation of left-side driving working for the networks in TRANSIMS –

a) Testing on parameters of routing
b) Flipped network analysis in TRANSIMS
c) Analysis of changes to be made in source code

.Though some of the experiments posed convincing results, other approaches still need to be further studies on. Also the calibration and all for the approaches needs to be done.

5.2 Changing parameters of Routing

Router utility as explained in chapter-3 is one of the important utility which decides the routes of vehicles based on dynamic traffic assignment techniques. Basis of deciding the routes lies in the way travel cost is calculated for each of the route while preparing for next iteration using outputs from last iteration. List of different input parameter that decide the way travel cost is calculated in TRANSIMS is shown in Table-5.1

Table 5.1 Travel cost Parameters

Parameter	Range(Default Value)	Definition
WALK_TIME_VALUE	0 to 1000 (20)	This value is multiplied to the total time spent on process links and walking on network, and added to the travel cost
BICYCLE_TIME_VALU	0 to 1000 (15)	Multiplied by total time on Bike, and

E		gets added to travel cost
FIRST_WAIT_VALUE & TRANSFER_WAIT_VALUE	0 to 1000 (20)	Relates to transit movement. Multiplies time spent waiting in transfer etc and adds to cost
VEHICLE_TIME_VALUE	0 to 1000(10)	This multiplies to the travel time on the link during time of day when vehicle would enter that link, and adds on to travel cost
DISTANCE_VALUE	0 to 1000(0)	If we need to give high importance to shortest path by distance, we give this parameters to multiply to total link length and adds on to cost
COST_VALUE	0 to 1000(0)	This value is multiplied by the cost value on Process Links, parking lots, and the transit fare
LEFT_TURN_PENALTY	0 to 10000(0)	For every left-turn it adds this value to the travel cost of the route
RIGHT_TURN_PENALTY	0 to 10000(0)	For every right-turn it adds this value to the travel cost of the route

As defined these parameters are defined with the router utility as custom parameters for deciding better route. What bring our special attention are the parameters LEFT_TURN_PENALTY and RIGHT_TURN_PENALTY which are defined as above in Table 1.

By definition if we try to understand how these would help towards simulating left-side traffic - Travel time calculation mainly forms the part of the feedback mechanism where when we decide routes for next iteration we compare the impedance value based on travel time on different routes. Now those parameters 1 & 2, serve exactly the same purpose.

For e.g. consider the network shown in Fig.5.1

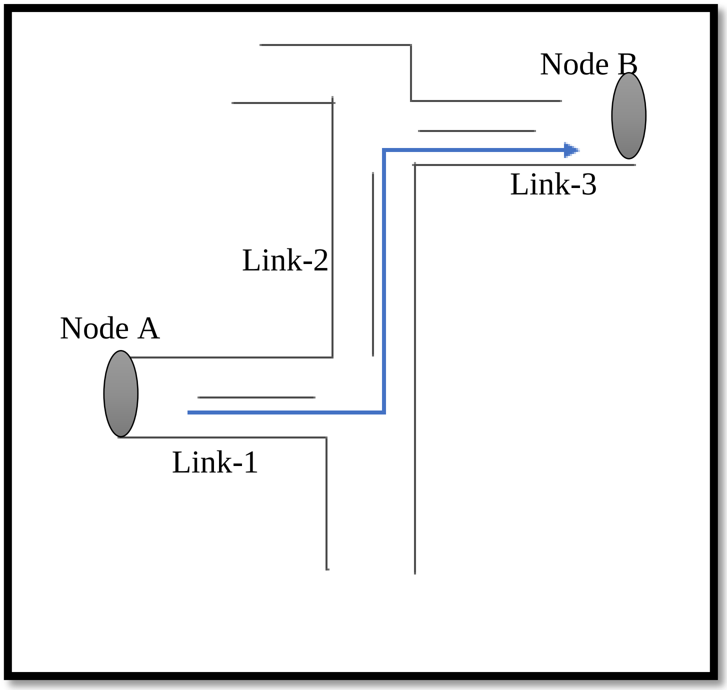


Fig. 5.1 Routing decisions

In a simulation for the network, travel cost for the route highlighted would involve multiplying the travel time on the route to the appropriate cost associated with that, also taking into account the congestion level on the links. Here the route involves an initial right turn, then a left turn, with travel time calculation for the route involving time taken to traverse both the turns.

So an effective travel cost calculation for the route shown would be like –

$$Cost_{r1} = T * v_t + d_v * L + C_v * F + L_p * n_l + R_p * n_r \tag{5.1}$$

where $Cost_{r1}$ as total travel cost on route r1 shown, where total travel time spent in vehicle is T and v_t is the VEHICLE_TIME_VALUE parameter, L is sum of travel distance on each link multiplied with d_v, the Distance_value. F refers to the total fare cost multiplied by COST_VALUE. L_p and R_p are left and right turn penalties multiplied by no. of left and right turns.

Now what we accomplish out of parameters LEFT_TURN_PENALTY and RIGHT_TURN_PENALTY is that by defining them to have a suitable value we can somehow add to the travel cost same amount of impedance that would have occurred if left-side driving would have happened.

For a proper justification of this, experiment was conducted on 2 networks that showed us the influence of these parameters on routing.

5.2.1 C-Shaped network analysis

A sample network shown in Fig.5.2 was fed as input to TRANSIMS

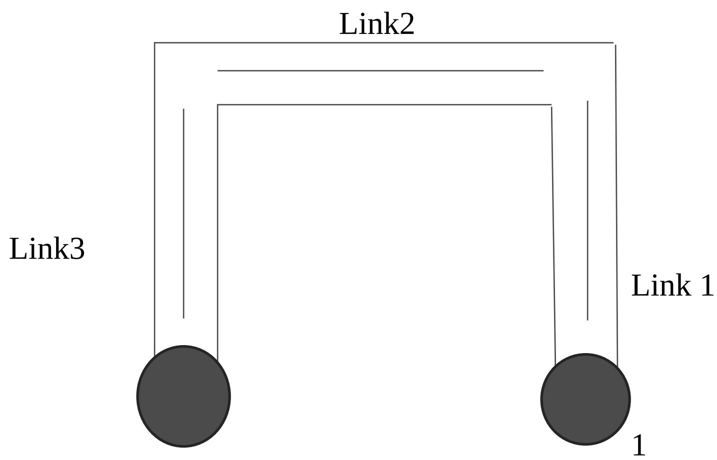


Fig. 5.2 Sample Network

The network was operated for following settings –

a) A simple OD matrix demand of 100 trips from 1 to 2 and 100 trips from 2 to 1, were fed inside the network

b) The trips were distributed in equal proportion between 7am and 8am (25% to each of 15min- time intervals) to make it a 1-hour simulation

c) Parameter RIGHT_TURN_PENALTY was set to 5000, while LEFT_TURN_PENALTY was set to its default value of 0

The output as obtained are being presented in the tables shown –

Table 5.2 C-Network output analysis

	Avg Time (sec)				Vehicle ID	TravelTime(sec)
Time Range	On Link1 (towards link 2)	On Link3(towards link 2)	vehicles from 1 to 3		1	76
					2	76
6:45 - 7:00	35.4	35.7			3	67
7:00 - 7:15	37.2	37				
7:15 - 7:30	40.8	41.1	vehicle from 3 to 1		2401	78

7:30 - 7:45	39.9	42.1			2402	78
7:45 - 8:00	42.1	40.3			2403	69

Observations from above state that –

Travel time taken by the vehicles on both the routes 1-2-3-4 and route 4-3-2-1 is approximately same, so is the average time taken by vehicles on link 1 and 3.

1-2-3-4 route has 2 left turns while route 4-3-2-1 has 2 right turns.

Conclusions that can be drawn from the above results are –

a) The parameters LEFT & RIGHT penalties in router file do not directly influence the travel time of the vehicles. Even after increasing or decreasing the right turn penalty to 10000 and 2500, no change in travel time was observed for individual vehicles.
b) Vehicles on both the route were found to have closer time, rather right turning vehicles were found to have slight more time (large by 2-3 sec approximately 4% of the time to travel), however this increase is not attributable to more value of RIGHT_TURN_PENALTY as it was found not to vary with changes in RIGHT_TURN_PENALTY.

5.2.2 C and inverted-C network

5.2.2.1 Network Details

This time a different network was fed into TRANSIMS. The characteristics and parameters of the network were-

a) There were 2-zones and traffic flow was from zone-1 to zone-2. The network was designed keeping in mind that TRANSIMS would get an option to decide the route for the vehicles travelling from zone-1 to zone 2
b) The length and turn characteristics of each of the route were kept same, except that one route involved 2-right turns(here from called the 'top route') and the other involved 2-left turns(here from called the bottom-route)
c) RIGHT_TURN_PENALTY was kept as 10000, whereas LEFT_TURN_PENALTY was kept to its default value 0.

The C-Inverted C network as observed is as shown in Fig.5.3-

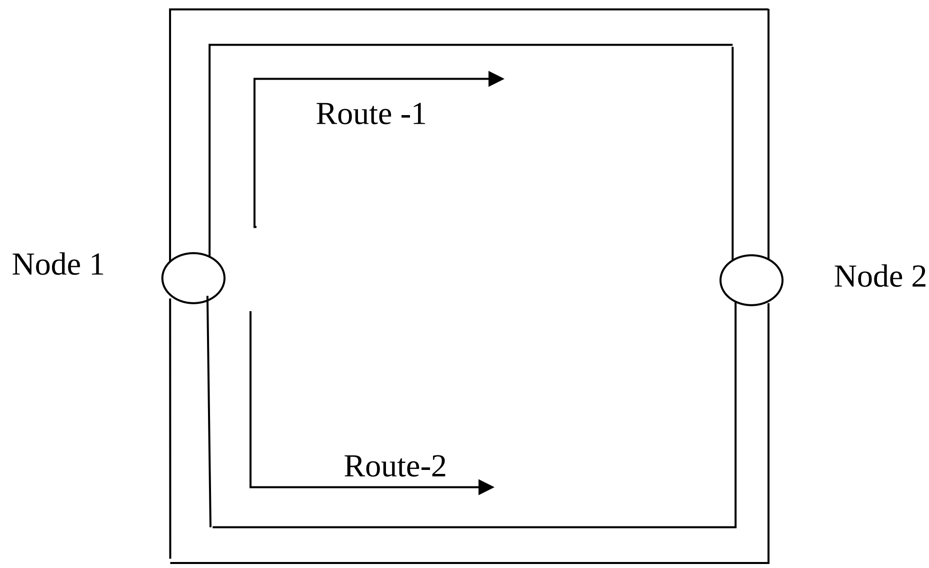


Fig 5.3 C and inverted-C network

The differentiating output obtained from simulation is as follows –

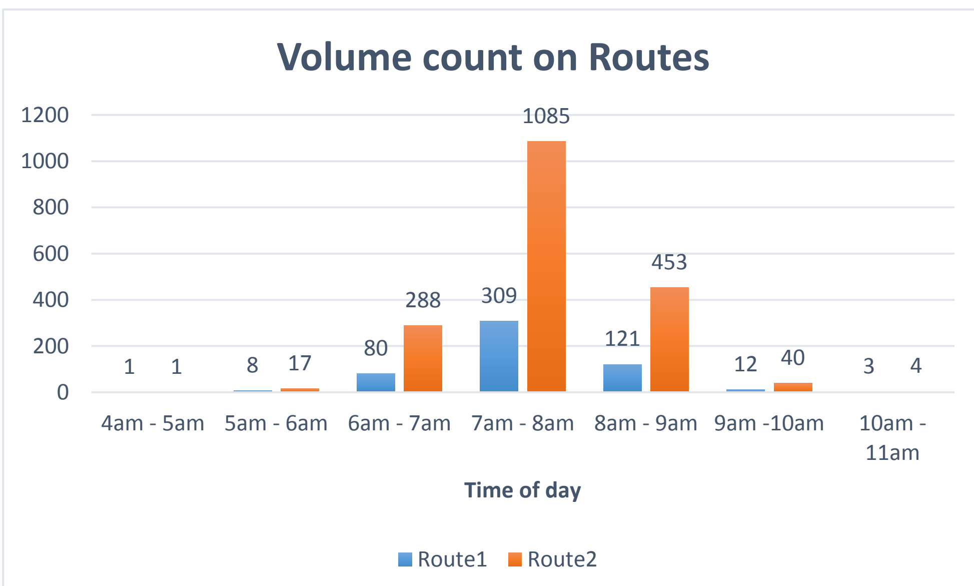


Fig 5.4 Volume count on Routes

5.2.2.2 Observations

a) Most of the vehicles choose the route with left turn and thus choose the route 2 with only left turns.
b) Some vehicles still choose path with right turn

c) Vehicles depart from origin to destination even as early as 4am, even when simulation period was specified as 7am-8am.

5.2.3 Conclusions drawn

Conclusions drawn from above outputs are as below –

a) Vehicles choose path with least travel cost. The parameters LEFT and RIGHT turn penalties, are into effect only when routing decision needs to take place, and are added to the travel cost on different routes, which in next time-step can influence choice of users.
b) Some vehicles still choose path with right turn, because of congestion on other route, and to ensure that dynamic user equilibrium satisfies.
c) As observed some vehicles started their journey way back from specified duration 7am-8am due to routing decisions which in order to improve travel times of users can to some extent change the departure times of the vehicles

The output observed from above two cases give us a major hint on the work that those parameters indeed influence the way we want the differences in left and right driving to reflect, i.e. when we want travel times for left-turning vehicles to be small in left-driving scenario at an intersection so that for next iteration we count less disutility due to travelling, we are in-fact implementing the same lessening in disutility by making LEFT_TURN_PENALTY lower in value than RIGHT_TURN_PENALTY. So these parameters are not influencing the travel times and the travel time values would still be inaccurate for vehicles. But we get out of these parameters the required disutility due to the accurate travel times and so our iterations make us converge to the correct equilibrium.

Though the major question now is how much should be the difference between the two parameters and what precise values would actually reflect the exact travel time conditions. The answer to this constitutes the calibration of the model to make it suit the realistic data from the field.

5.3 Flipping the network

5.3.1 Concept

As had been described with the logic, basically flipping the network can result in desired changes in the network. To flip the network means to take the mirror image of the network about y-axis and effectively change the x-coordinates of the nodes to their respective negatives. Under this experiment a typical 4-arm intersection was simulated once with usual travel directions and then with flipped directions to analyze the differences in the outputs (link-volume and link- travel time). Fig.5.5 shows a sample box network being analyzed under flipping and usual case

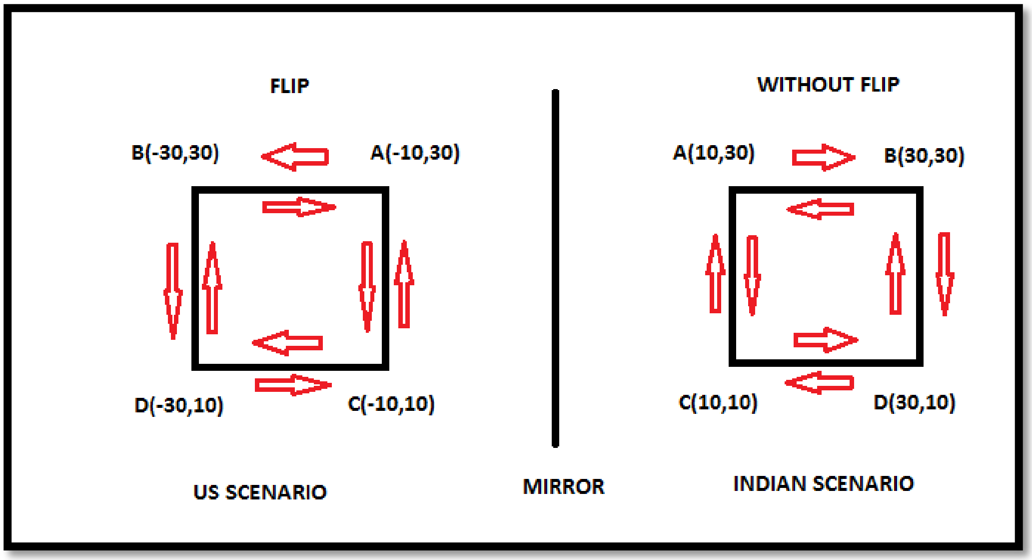


Figure 5.5 Mirror image of a right-side driving network

It clearly shows that when taken a mirror image, TRANSIMS will implement the intended left-side driving on original network, even while implementing the inherent right-side driving on the flipped network. Thus for any network the outcomes of a flipped network will be the outcomes of the original network of the corresponding links.

For an instance from above reference (Fig.5.5), we say when we model network shown on right of the figure by in fact modeling the network shown on left, in TRANSIMS, we will get the flow values of each link very identical to what is expected, i.e. flow from A to B on left network will be same as flow from A to B in right network. Such correlation matches perfectly and even while analysis of outputs if we correspond to same node and link-id's we'll get the desired results for left-side driving without doing anything to the outputs.

Such flipping was tested on a sample 4-arm intersection modelled in TRANSIMS. The three instances in Fig.5.6 show the sample intersection network and explains how we implement the same logic by flipping of the network about y-axis. If we say 50 trips are made from 3 to 1, then as shown in the figure we'll have same travel time for those 50 trips in network implementations shown in Fig.5.6 (a) and Fig.5.6 (c) because of similar behavioral trend of vehicles in both networks.

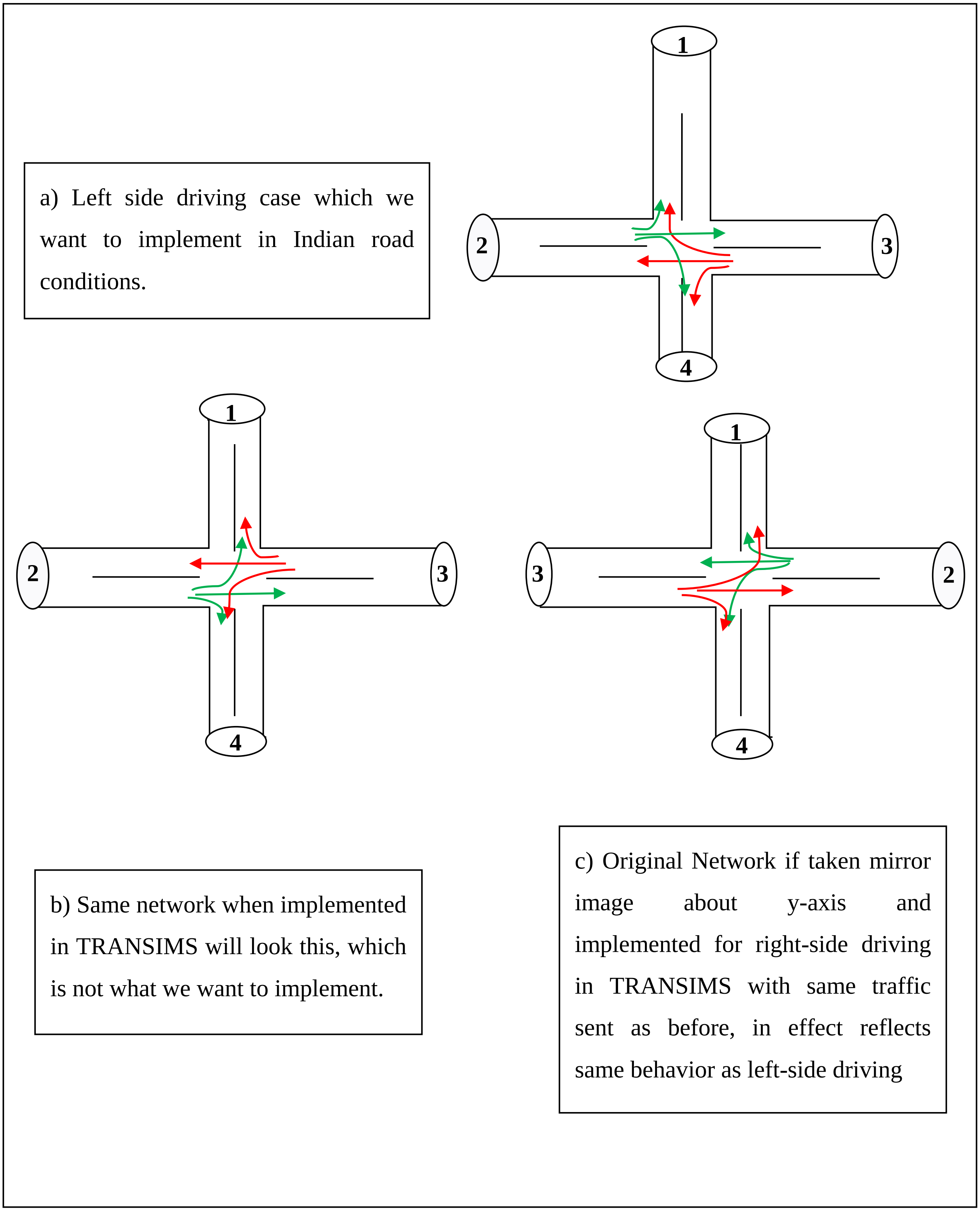


Fig.5.6 – Flipping algorithm to model left-side driving

Also note that link volume observed on link connecting 2 to intersection node will be same in Fig.5.6 (a) & Fig.5.6 (c), and similarly for all other links, thus similar analysis would hold.

5.3.2 Observations

Same analysis was applied for the later 2 networks (Fig.5.6 (b) & Fig.5.6 (c)) in TRANSIMS and following results were obtained –

Table 5.3 (a) Volume and Travel Time for movement from 1 to 2

				Right direction movement(Fig.5.6 (b))		Left direction movement(Fig.5.6 (c))	
Start Node	TimeStart	Timeend	End Node	No. of vehicles taking the turn	Avg Time Taken on inLink before turning	No. of vehicles taking the turn	Avg Time Taken on inLink before turning
1	7:15	7:30	2	41	47.5	56	51.7
1	7:30	7:45	2	45	50.7	57	55.8
1	7:45	8:00	2	42	48.1	51	53.3
1	8:00	8:15	2	32	40.6	39	44.5
1	8:15	8:30	2	20	30.7	24	32.5
1	8:30	8:45	2	11	21.9	15	21.9

Table 5.3 (b) Volume and Travel Time for movement from 1 to 3

				Right direction movement(Fig.5.6 (b))		Left direction movement(Fig.5.6 (c))	
Start Node	TimeStart	Timeend	End Node	No. of vehicles taking the turn	Avg Time Taken on inLink before turning	No. of vehicles taking the turn	Avg Time Taken on inLink before turning
1	7:15	7:30	3	61	62.9	59	34.5
1	7:30	7:45	3	66	68	62	37.5
1	7:45	8:00	3	63	64.7	54	36.1
1	8:00	8:15	3	51	53.8	39	30.6
1	8:15	8:30	3	34	39.5	23	23.1
1	8:30	8:45	3	20	27	11	16.8

Do note that the travel time estimation shown are the average travel time taken on the link connecting node1 to intersection which includes the waiting time at intersection for signal in that direction to clear up, but doesn't include the time taken by vehicle to traverse the intersection as the link has already been left once it enters the intersection.

5.3.3 Conclusions drawn

Following observations and conclusions can be drawn from the tables above -

a) For same direction movement a left turn (from 1 to 3) causes more travel time in a right sided traffic movement as compared to left-sided traffic movement. This is solely because of the signal plans that give preference to the easy movement

b) The observations for network in Fig.5.6 (c) indeed reflect the same manner in which a left side driving scenario would implement the movement of traffic.

c) Slightly lesser travel times were observed for left-side driving scenario, but this is not attributable to the nature of modelling done by TRANSIMS, rather might be because of the input OD matrix which might have caused different congestion levels on route 1 to 3 and route 1 to 2, in both the networks.

An attempt was also made to compare this left-side driving simulation results with the VISSIM simulation however due to inability of TRANSIMS to specify width of the lanes didn't quite result in match of simulation outputs.

Chapter-6 Track-1 re-implementation for IITB Campus

6.1 General

Once the formal structure of the procedure to be applied to transims was found clear, applying the understanding developed on a real-life model was the need. Saxena(2012) mentions about implementing the IIT-Bombay Track-1 simulation. However the results of the calibration were not precise and even there were several unrealistic unsorted errors which could have changed the simulation results directly. Thus a refined attempt was made at building the Track-1 implementation of the IIT-Bombay campus. Crucial efforts were put to align the values of the parameters used at several different steps to real life Indian Traffic conditions.

6.2 Network Preparation

One of the most crucial steps in building a simulation is ensuring that network coding happens exactly the way we want it to be. Several documents like California department of transportation guidelines on travel modelling, stress on high importance of this. For IIT-Bombay campus, the previous studies were a base.

Coding of the network was established to be correct by checking for each of following –

6.2.1 Links, Nodes and Zones

As explained in Sec.5.3 Saxena(2013), GISNet.exe was used to build node and link files from the shape files extracted from the OpenStreetMaps. GIS net gave a generated link, node and zone file which had several dangling errors which were manually sorted. Link file was also updated manually, to remove two links in opposite directions, and were rather made one link with lanes in opposite direction available. This was done to get realistic model, as in TRANSIMS if a link has lane available in other direction too then it is modelled as undivided street, where vehicle travelling in other direction can come on this side lane to overtake.

Zones considered were similar to one used by Kanchana(2011) and Saxena(2013). An extra addition was made to the zones, by creating an extra zone 11 near YP gate. This was done in accordance with the concept that the trips can exit our zone from these two external points. Alexandria was the base case from which the idea was taken.

Several of Link and zone attributes were changing their values when passed through TransimsNet utility. Thus several manual changes were done after execution TransimsNet, in the new Link,Node and Zone file written by it, for instance changing the use code of external links from ‘WALK’ to ‘ANY’ formed one of the major change.

6.2.2. Activity locations & Parking lot

Fig.6.1 shows the activity locations from where the trip would originate and end, created out of TransimsNet utility.

Figure 6.1 Activity Locations zone wise

The paper by Shan Huang et al. (2012) talks about manually moving these activity locations to precisely the point where the trips end or start. This was however not done for our network. Again because Transims generated activity locations were more or less the position where trips can end or originate as there are no specific parking lot at several locations in the campus. There were no activity location generated for walkways which goes as per definition of walkways. Following parameters were used in TransimsNet control file to get activity locations as shown –

```
ACTIVITY_LOCATION_SIDE_OFFSET          3              //---- meters ----
```

```
MAXIMUM_ACCESS_POINTS                3

MINIMUM_SPLIT_LENGTHS                60, 60, 60, 60, 60, 60, 60, 60   //---- meters
```

Every link could have a maximum of 3 access points and distances between each were also to be more than 60m. Parking lots and process links were attached right at the place where activity location was put.

6.2.3 Lane connectivity

Every link should have the connection with other links as desired for the network. For IIT Bombay campus every link allows for U-Turn at its end and even somewhere in between (which can be modelled by breaking links into several smaller links which though is again dangerous for Transims to simulate). Then we have for every intersection the suitable connection for movement in all directions. Following Lane connectivity parameters were used in TransimsNet control file –

```
MAXIMUM_CONNECTION_ANGLE          180

ADD_UTURN_TO_DEADEND_LINKS              YES
```

Maximum value of connection angle was used because inside campus practically movement can happen at any degree deviation from original direction. The lane-connectivity at a typical intersection looked like this –

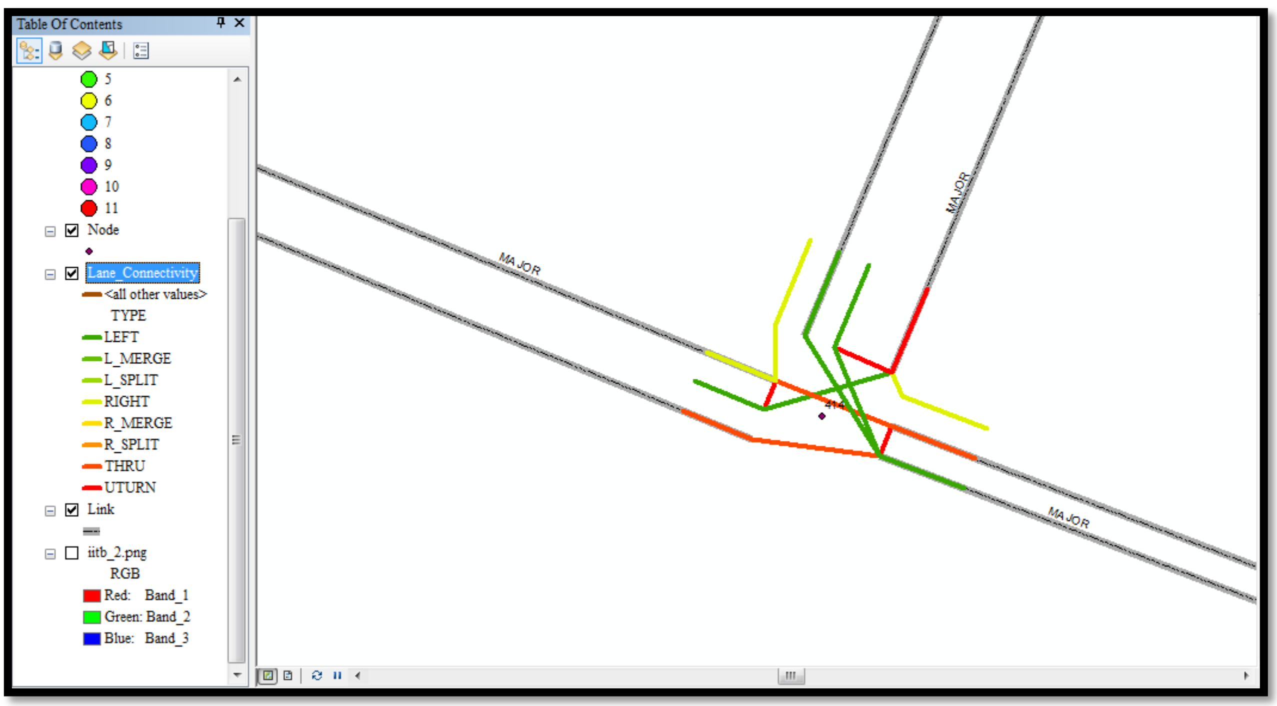


Figure 6.2 - Lane connectivity at NCC intersection

6.2.4 Signal controls

Since most the intersections inside IIT campus were unsignalized, no signal has been put at any node inside the network. However for the 2 external links we did put 2 signals with cycle length approximately 120sec and green times allocated in proportion to capacities. This was done only to simulate the signalized movement at main-gate and YP gate and may not represent the true value of cycle and green time lengths at those points. But this was found to be not much material in regards to simulation output as a vehicle once reaching the main-gate exit link is technically out of network. At other nodes inside the network sign control were put up, which ask for vehicles on lower priority link to yield or stop before the vehicle on high priority links. Again since most of the link types inside the campus are identical this priority rule doesn't have much effect and simulates the real behavior inside the campus. Following snapshot shows the presence of Timed and Sign signals put up at different nodes –

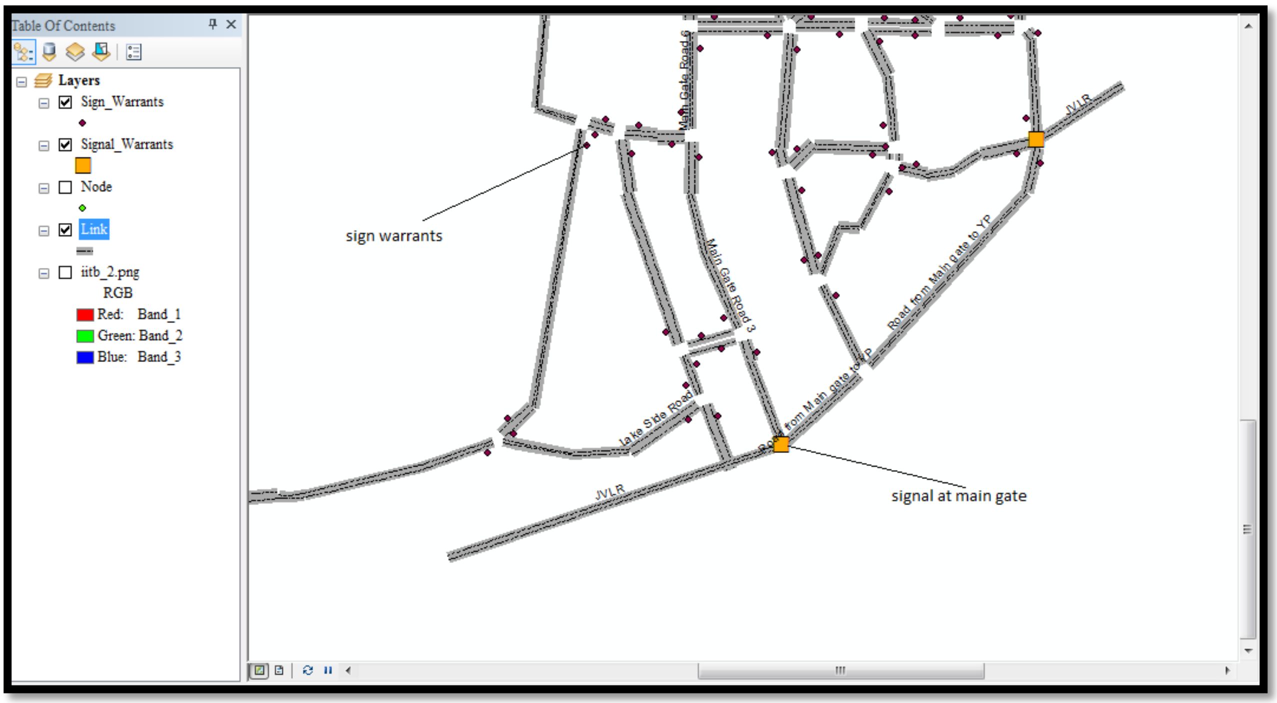


Figure 6.3 Signals and signs at different nodes

The parameter values used for preparing signal were following-

```
SIGNAL_CYCLE_LENGTH            120         //---- seconds ----
MINIMUM_PHASE_TIME             5           //---- seconds ----
YELLOW_PHASE_TIME              3           //---- seconds ----
RED_CLEAR_PHASE_TIME           1           //---- seconds ----
```

```
SIGNAL_SPLIT_METHOD                     CAPACITY
MINIMUM_LANE_CAPACITY                   500          //---- vehicles / lane / hour ----
MAXIMUM_LANE_CAPACITY                   1500        //---- vehicles / lane / hour -----
```

, where each parameters explains for self on how it was used to decide green time and phases for the signal placed at main gate and YP gate.

6.2.5 Transit routes and schedules

The routes of the buses were sorted using CUBE (Saxena, 2013). However as compared to network used by Kanchana(2011), only three of the prominent bus routes were implemented in the network, as the route coding for others were problematic and that the trips made on those routes for the comparison time period were also very less. For the routes that were incorporated namely lake side route, 2A bus route and hill side bus route, following information were provided-

```
STOP_SPACING_BY_AREATYPE            100, 150, 200      //---- meters ----
TRANSIT_TIME_PERIODS                5:00, 7:00, 9:00, 15:00, 17:00, 20:00 //
TRANSIT_TRAVEL_TIME_FACTOR          1.0, 1.1, 1.22, 1.1, 1.25, 1.1, 1.0
MINIMUM_DWELL_TIME                  5            //---- seconds ----
```

Transit stops were manually coded to the position of exact stop on the network. Then with the RouteHeader file(as shown in Table 6.1), headway for buses on each route was provided as 10 minutes during peak hour time bins as specified in TRANSIT_TIME_PERIOD field, with the offset value from start of time bin being randomly generated since each of OFFSET_x is set to a negative value. For eg. Headway_3 refers to time-bin 7:00am-9:00am and frequency of buses would be 10min in it.

Table 6.1 Sample Route header input

ROUTE	NAME	MODE	HEADWAY_1	HEADWAY_2	HEADWAY_3	OFFSET_1	OFFSET_2	OFFSET_3
1	R1	BUS	0	0	10	-1	-1	-1

Following picture depicts the 3-routes in the network –

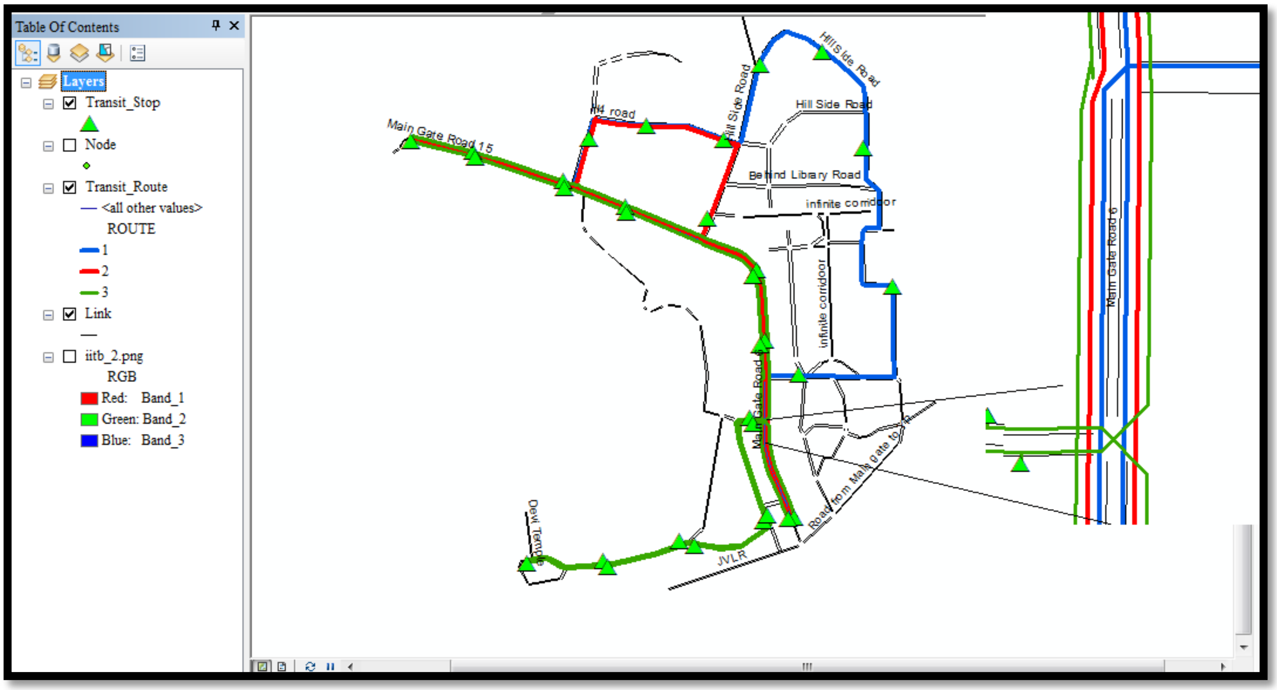


Figure 6.4 Transit stops and Routes in the network

After establishing that all the network elements were in place further processes were carried ahead. Input shape file was however not provided, which could have been done for the better shape of the links.

6.3 Providing Trip matrices

Next step was to load the pre-defined origin-destination matrix over the network. The way TRANSIMS does execution of zone-to-zone OD matrix is to allocate each trip for a given OD pair to one randomly selected activity location in Origin zone and then selecting based on inverse weightage of distance(or weightage value provided in Activity location file) an activity location in destination zone. Thus an 11x11 OD matrix reduces to NxN matrix where N is the no. of activity locations in the network.

The source of the OD matrices was the survey conducted by Kanchana(2011). The Transims model thus models the IIT-Bombay network for the demand and infrastructure in 2010. The survey conducted was for 2 hours from which OD matrix of 1.5 hours was obtained as below.

Table 6.2 Original OD table

		D	E	S	T	I	N	A	T	I	O	N	S
		1	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**	**10**	
	1	15	10	14	52	21	16	7	8	21	47	**211**
O	**2**	16	2	17	44	11	15	2	2	2	81	**192**
R	**3**	16	9	0	8	3	5	1	0	2	18	**62**
I	**4**	64	72	36	50	37	76	7	19	20	157	**538**
G	**5**	28	8	16	21	0	6	1	1	5	31	**117**
I	**6**	11	7	6	51	6	1	11	7	1	66	**167**
N	**7**	6	1	0	5	0	12	2	5	5	14	**50**
S	**8**	9	1	5	21	3	7	5	1	12	16	**80**
	9	18	6	2	22	3	5	4	11	0	45	**116**
	10	37	51	23	93	24	68	13	15	45	18	**387**
		220	**167**	**119**	**367**	**108**	**211**	**53**	**69**	**113**	**493**	

To extrapolate it to 24hours trip demand we used following assumptions-

a) 70% of the trips are HBW while 30% are HBO. This assumption doesn't hold for LCV, HCV.

b) Peak hour Volume = 10% of Daily Volume

c) All the trips originating or terminating in zone 10 were further split into coming from zone10 and zone11, assuming 70% of those trips come from Maingate and 30% of trips come from YP gate.

Vehicle composition used was-

Table 6.3 – Vehicle composition

Vehicle Type	% Composition
Car	23.45
Motorbike	52.44
Auto	17.33
HGV	2.77
LCV	2.77
Tum-Tum	1.24

Hence then simple linear exploration was assumed and trips throughout the day were given a picture. For eg. Total no of home based work (HBW) car trips from zone 1 to zone 10 throughout the day are given as –

Total no of trips to zone10 and 11= 0.7*0.2345* (100/10*14) = 22.98 = 23

Total no. of trips to zone 10 = 0.7*22.98 = 16.1 = 16

Each of the trips was rounded to the nearest whole number because transims models every individual at a very discrete level.

Now to frame trip tables following approach was used –

a) Separate trip table was written for each of vehicle, so that easy manipulation can be done later, and also because transims asks for trip table of every vehicle individually. Above extrapolation logic was used.
b) Diurnal distribution was created keeping in mind that from 4pm to 6pm, total of twice of no. of trips happening in peak observed hour, happen in those 2 hours. Thus the diurnal contained values only from 4pm-6pm and rest were kept zero. (16.3 means 16hours + 0.3*60min = 16 hours 18 min i.e. 04:18pm)

Table 6.4 Sample diurnal distribution

START	END	SHARE
15.5	15.8	0
15.8	16	0
16	16.3	0.14
16.3	16.5	0.11
16.5	16.8	0.14
16.8	17	0.11
17	17.3	0.14
17.3	17.5	0.11

c) Separate AP and PA trip files were created, where AP stands for Attraction-production and PA for production-attraction. The difference is based on the value of the key TRIP-CONTROL_POINT for both the tables. The diurnal distribution produced by the trip time file is used to define the probability that a given trip will take place at a given time. The time assigned to a trip is based on the offset of a random number within the cumulative probability distribution. By default, this time value (in seconds) is assumed to be the time at the origin of the trip. Thus AP trip table is applied with TRIP_CONTROL_POINT key value as destination, meaning that the diurnal distribution is an arrival time distribution, while for PA trip table diurnal distribution is departure time distribution. Values in both the tables were the same and thus a trip factor of 0.08 was applied to both the trip table definition in control file of convert trips to account for not counting the trips twice and to get twice the peak hour flow from 4pm-6pm.

Under the control file of convertTrips, two paramaters were used namely –

```
ADDITIONAL_TRAVEL_TIME          120          //---- seconds ----
RANDOM_NUMBER_SEED              121345
```

,where additional travel time was added to every trip to account for losses in getting into vehicle and other random delays. The random seeds helps in random choice of activity and parking locations in the zone. The vehicle_type file given as input to convertTrips program contained the crucial vehicular characteristics. A sample is as shown –

Table 6.5 Vehicle type file input

TYPE	SUBTYPE	LENGTH	MAX_SPEED	MAX_ACCEL	MAX_DECEL	USE	CAPACITY	LOADING	UNLOADING	METHOD	MIN_DWELL	MAX_DWELL
1	0	7.5	35	7.5	7.5	CAR	2	3	2	PARALLEL	0	0
1	1	3.5	27.78	3.2	3.2	BIKE	1	0.1	0.1	PARALLEL	0	0

The values used for each of the vehicular parameter were taken as per Indian standards. The average velocity for each of the mode was specified in trip file and was also based on realistic Indian traffic conditions.

6.4 Traffic Assignment

Routing also makes one of the most crucial steps in making a discrete event traffic simulation possible. As learnt before several iteration of its are used to make the best plan for the user under which they go in dynamic-user-equilibrium (DUE), which is then assumed to model the real traffic very closely. Much of the time was spent in making sure that every step of routing iteration produces the desired result. Total of 10 iterations were used to make the travel behaviour converge to dynamic user equilibrium.

6.4.1 Setting up parameters - first 5 iterations

The first five router iterations used the principle of evaluating the experienced link travel times from the travel time function of the link (typically the BPR function). This link travel times were calculated from the PLanSum utility instead of doing the actual microsimulation. The process setting up critical parameters for each of the utility-

6.4.1.1 Router.exe

Under it the typical parameters of routing were set. Indeed there was a need to choose suitable value of those parameters for best calibrated results, yet in start some typical default values were chosen. The penalty values that are used in generalized cost function set by the router were set. The approximate contribution of each penalty parameter was also judged –

Table 6.6 Parameters used for Routing

WALK_SPEED	1.2
WALK_TIME_VALUE	0
VEHICLE_TIME_VALUE	200
FIRST_WAIT_VALUE	125
TRANSFER_WAIT_VALUE	20
DISTANCE_VALUE	1
COST_VALUE	5
TRANSFER_PENALTY	1200
MAX_WALK_DISTANCE	3500
MIN_WAIT_TIME	60
LEFT_TURN_PENALTY	0
RIGHT_TURN_PENALTY	500
UTURN_PENALTY	2000

Zero value of WALK_TIME_VALUE was chosen so that more people should prefer direct walk mode. On similar logics higher value of vehicle time and walk time penalties were chosen. Cost value was not of much effect because transit trips inside campus are free for residents (or very minimal value to be accounted for). Maximum walk distance was kept as 3.5km to account for all long walk trips that can possibly happen. Some start values of turn penalties were also chosen.

What router generates is the travel plan of every individual from origin activity location to destination. The router errors were minimized to ensure that no less than 5% of the travelers should go un-routed. The prn file of the router iteration showed –

```
Percent of Total Trips with Problems = 1.7%

Total Number of Problems = 339

Number of Zero Node (#3) Problems = 182 (53.7%)

Number of Park-&-Ride Lot (#12) Problems = 157 (46.3%)
```

Most of the zero-node problems were external to external trips and thus were hardly of any concern. Park and ride trips also don't hold any relevance either. Figure 6.5 show the travel plan as developed for a traveler travelling from zone-1 to zone-4.

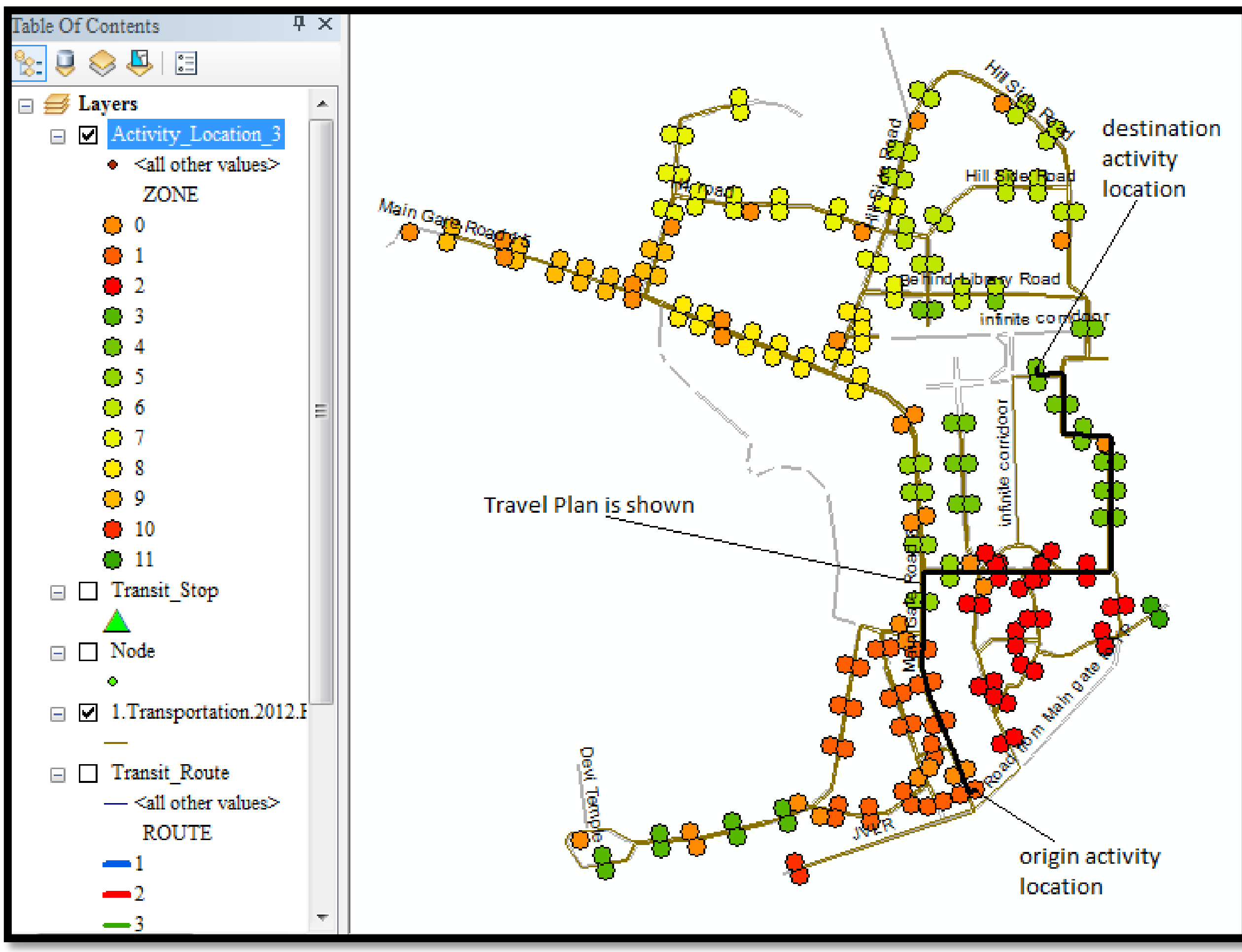


Figure 6.5 Travel plan of traveler 150761 from zone 1 to zone 4

6.4.1.2 PlanSum.exe

Under this utility TRANSIMS simulates the routes developed by previous utility and evaluate the experienced link travel times, using the BPR function. The parameters given to it were as given –

```
EQUATION_PARAMETERS_1                    BPR, 0.15, 4.0, 0.75
EQUATION_PARAMETERS_2                    BPR, 0.10, 4.5, 0.75
```

i.e. the function of travel time function in equation-1 is given by –

$$t = t_0(1 + 0.15 * (\frac{V}{0.75C})^4)$$

, which will then be evaluated using predicted traffic volume on the link in 15-min time bins as judged by the approximate volume in those bins. Later these improved experienced travel times will be used to evaluate new routes in next iteration.

It also helps in printing several useful reports in its .prn file like – Top_100_v/c_ratios, Top_100_link_volumes etc.

6.4.1.3 PlanSelect.exe

This utility doesn't run in 1st iteration, and from 2nd iteration it runs before the router utility and selects the households from the set of several households which need to be re-routed for next iteration. This selection is done of course to avoid swinging back and forth about equilibrium if all the travelers are allowed to re-route at same time. The parameters set under it are given as-

```
SELECT_TIME_PERIODS                15:00...18:30
SELECTION_PERCENTAGE                     80              //---- percent ----
MAXIMUM_PERCENT_SELECTED                 20              //---- percent ----
RANDOM_NUMBER_SEED                       255
```

For first 5-iterations we used selection criteria of routing the travelers which travel in time bin 3pm-6:30pm. This however just repeats the process for next 5-iteration. An improved version would involve selecting congested links which in this case were not available as actual to free flow time ratio was very close to 1 for all links.

6.4.1.4 PlanPrepare.exe and LinkDelay.exe

PlanPrepare would simply merge the newly routed travelers with the old travelers which were not selected for re-routing. Nothing much goes with its parameters else than good summing up reports like PERCENT_PATH_CHANGE and percentTravelTimeChange.

LinkDelay does very much the conversion process of files generated as output to the PlanSum and make them into smoothened ‘travel time by time of day’ file for use by further iteration. Its parameter tell about how to go about smoothing wrt different travel times for the same link–

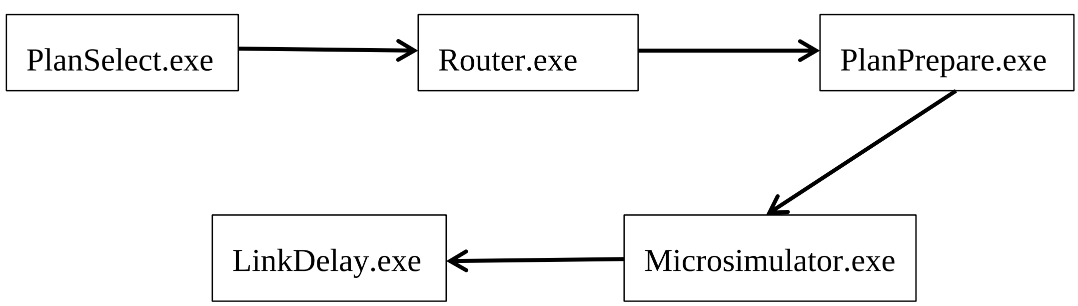
```
SMOOTH_GROUP_SIZE                       3
PERCENT_MOVED_FORWARD                   20
PERCENT_MOVED_BACKWARD                  20
```

It does the weighted average of window of size 3, by moving 20% backward and forward for the data.

6.4.2 Setting up parameters – next iterations

Next 5 iterations involve real micro-simulation of all the vehicles on the network as per their travel plans. There the role of microsimulator and real interaction between different vehicles comes into picture and this effectively brings a better measure of link volume and realistic experienced travel times.

The process and setting of parameters hold the very same here too with the order of execution of utilities being -



6.4.2.1Microsimulator.exe

Under this utility lie the most crucial parameters of simulation which relate with the movement of vehicle on the celled links. The typical setting of parameters for this involved looking at the usual real-time vehicle movement settings which involve following –

Table 6.7 Microsimulator Parameters used

CELL_SIZE	7.5mts
TIME_STEPS_PER_SECOND	1 steps/sec
TIME_OF_DAY_FORMAT	24_HOUR_CLOCK

SIMULATION_START_TIME	0:00
SIMULATION_END_TIME	25:00:00
SPEED_CALCULATION_METHOD	CELL-BASED
DRIVER_REACTION_TIME	0.7,0.8,0.9,1
PERMISSION_PROBABILITY	55%
SLOW_DOWN_PROBABILITY	10,15,20
SLOW_DOWN_PERCENTAGE	10,20
RANDOM_NUMBER_SEED	1623
PLAN_FOLLOWING_DISTANCE	525mts
LOOK_AHEAD_DISTANCE	260mts
LOOK_AHEAD_TIME_FACTOR	1 imped/second
LOOK_AHEAD_LANE_FACTOR	4imped/laneChange
MAXIMUM_SWAPPING_SPEED	37.5mts/second
MAXIMUM_SPEED_DIFFERENCE	7.5mts/second
ENFORCE_PARKING_LANES	Yes
MINIMUM_WAITING_TIME	60sec
MAXIMUM_WAITING_TIME	120sec
MAX_ARRIVAL_TIME_VARIANCE	60min
MAX_DEPARTURE_TIME_VARIANCE	60min

These parameters over represent the behavior of traffic as expected in a university campus. Some values were taken from Alexandria default case and some were modified to approximately suit Indian traffic conditions.

6.5 Output Analysis

TRANSIMS builds down several outputs from its microsimulation and provides enough detailed and aggregate reports that are required for comparison and study of the region. Of typical analysis outputs were the arcgis plots of speed bins and link volume for every link and color coding the network based on that. ArcSnapshot and other Arcnet utilities also provide visualization for the snapshot of network at any point of time and also in determing the links with problems by overlaying/joining them with link attribute table. Knowledge of ArcGIS served very complimentarily at every step.

Visualizers were also use to get the feel of the real simulation being happening and identifying if any portion of network is not behaving the way it should. Both Nexta and TransVIS have been used for this purpose. Nexta also generates some interesting statistics namely measure of effectiveness, showing bottlenecks and shortest paths between nodes.

As per guidelines in Reference-9, animation should be used as a means to detect major behavior of the network and by building heat-plots and following a selected vehicle one definitely gets to know the problem behavior in the network. Following snapshots provide the glimpse of campus network simulation in TransVIS and Nexta.

Figure 6.6 Trans-VIS following selected vehicles and built heat plots

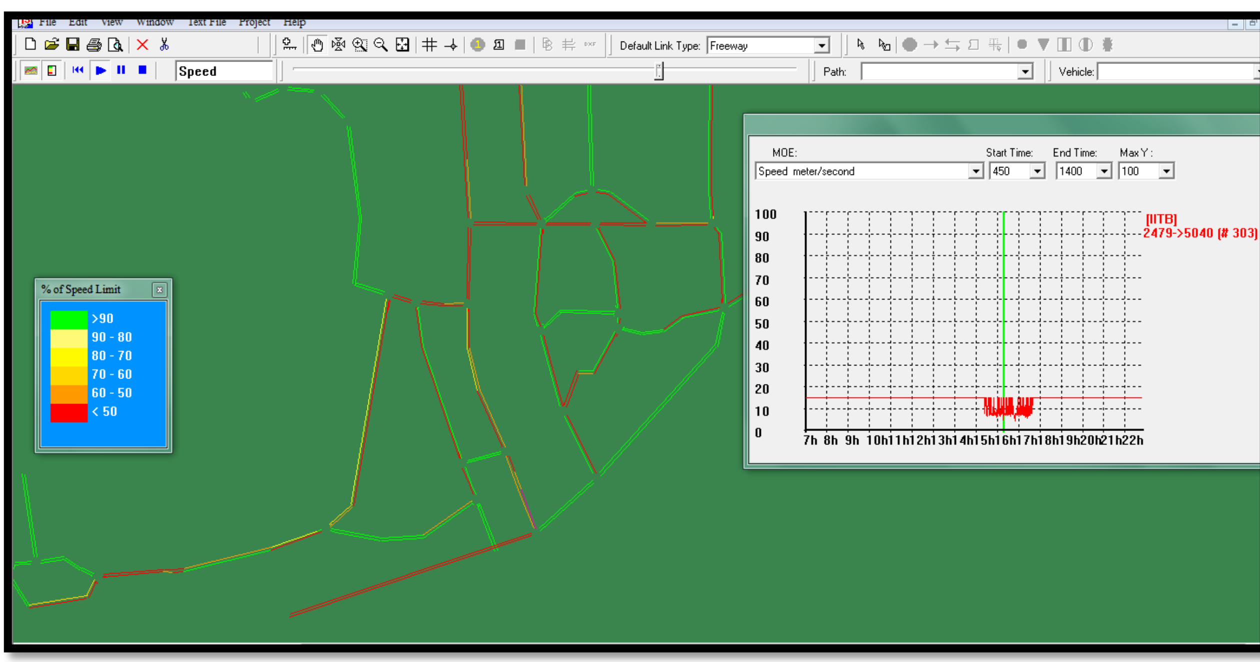


Figure 6.7 Display of speed-MOE-charts and linkwise speed info in NEXTA

However there is a degree to which the information from both the visualizers can be trusted. TransVIS being more realistic displays second by second movement of vehicle but does suffer from several operational drawbacks if heavy loading of network or several parametric changes are incorporated. Many of the options in version-4 visualizer are not yet enabled and suffer from several drawbacks.

Similarly in Nexta, one only gets counted values of Measures, which usually don't match with the calculated measures from TRANSIMS utilities. Major drawback with it being it shows minute to minute movement of traffic which doesn't quite serve any purpose of visualization, except for aggregate overview.

Typical chart of simulation time spent in executing the different utilities for IIT-Bombay campus were –

Table 6.8 Simulation time details

TransimsNet.exe	1.2 sec
IntControl.exe	0.8sec
TransitNet.exe	1.3 sec
ConvertTrips.exe	2.1 sec to read 4000 trips
5 Router iteration without Msim	3.5sec = 0.7sec/iteration
10 Router iteration with Msim	23sec = 2.3sec/iteration

However the time-spent in analyzing the output was considerable and consumed majority of efforts.

Chapter-7 Left-side driving and Calibration

7.1 General

To get left-side driving both the described methods were applied and tested. For calibration of the output data values collected in report of Kanchana (2011) were used. The data of traffic volume were observed at 16 different locations across the campus and of traffic speed at 10 different locations. To calibrate a model it was tested differently with different parameters, following the guidelines in the Guidelines by California department of transportation for travel modelling. Following link correspondence was chosen to compare volume and speed

Table 7.1 Link comparison chart

Traffic Volume		Speed	
Data Collection point	Link In Transims Network	Data Collection Point	Link in Transims Network
1	303	1	41
2	109	2	40
3	362	3	326
4	329	4	328
5	11	5	332
6	136	6	143
7	65	7	64
8	121		
9	337		
10	143		
11	325		
12	6		
13	2		
14	332		
15	40		
16	353		

7.2 Flipping the network

To simulate the flipped network, every x-coordinate in original node and zone file was multiplied with a negative sign and the result simulated. The modified network looked something like this-

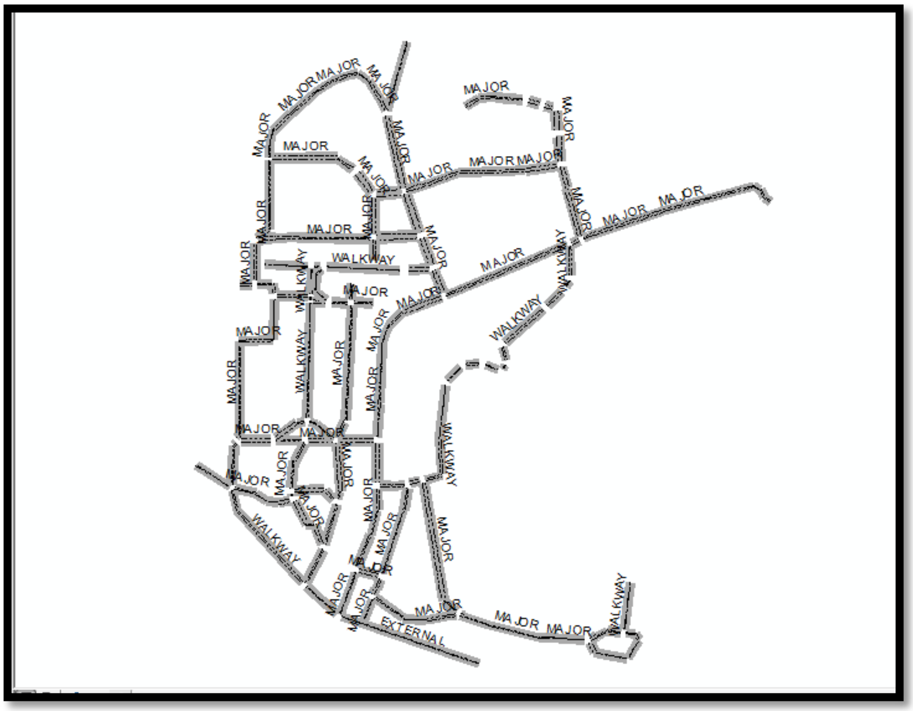


Figure 7.1 Flipped network

The parameters used for simulation were kept as identical as discussed before. The results of the simulation were compared with the original network simulation and observed field data. Fig.7.2 and Fig.7.3 present the percentage difference observed from field data

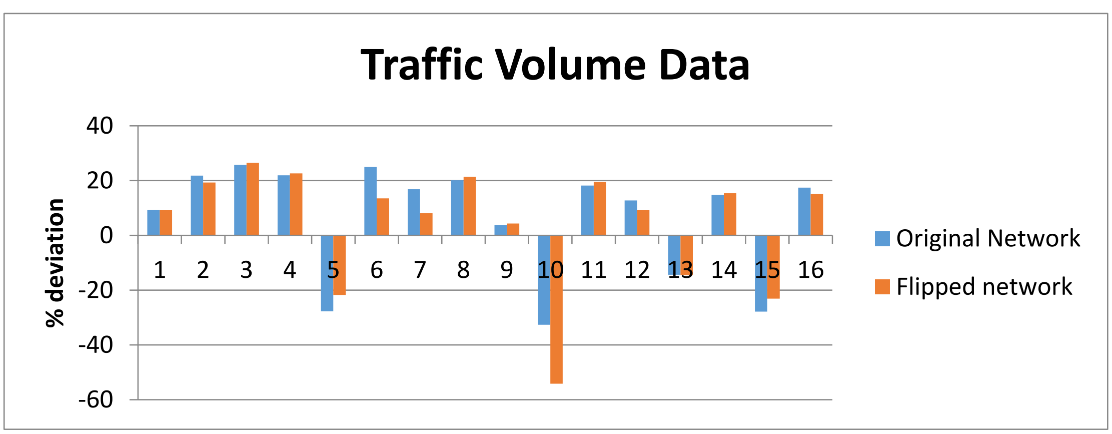


Figure 7.2 Percentage deviations in Traffic volume data

The table for this chart is attached in Appendix-C. As can be observed –

% deviations for both the cases are fairly identical. However value of deviations from original network is slightly higher expect at 2 locations. This can be explained from the fact that inside IIT campus since there is no signals, wait time for vehicles don't come into picture and thus the only difference between left and right side driving at intersection comes because of slight turning excess time.

Fairly closer values of observed traffic volume were observed implying closer level of simulation. However further calibration can be done.

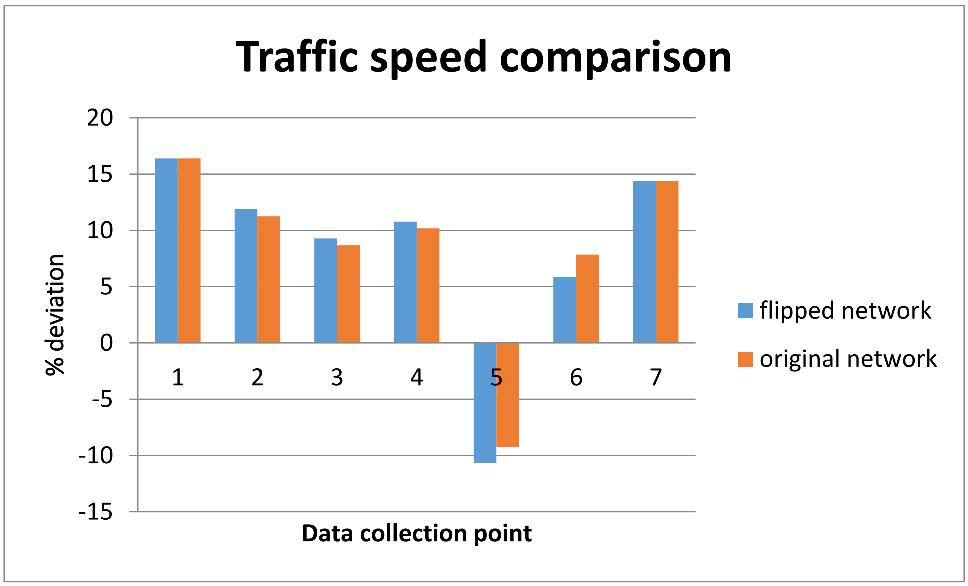


Figure 7.3 Percentage deviation in travel speeds

Similar behavior wrt traffic volume is observed. Speed values are more or less the same as original network. The data for the same is provided in the appendix. Other system variables as compared for original and flipped network as observed for last iteration are –

Table 7.2 Comparison chart

	Original Network	**Flipped network**
Total VHT	134.5 hours	134.9 hrs
Avg. Vehicle travel time	2.68minutes	2.69minutes
Total Vehicle Trips	3007	3005

Again the system variables are observed to be very close implying that in absence of signal both left and right turn movement behave identically. Both the cases were simulated for zero values of left and right turn penalties.

7.3 Calibration of Left and Right turn penalty parameters

Since the traffic speed data was giving almost similar results for different values of penalties, traffic volume response was used as the base to determine the optimum value of parameter for the least deviation from the observed.

Entire simulation from router utility was repeated for following values of parameters –

Simulation-1 leftTurnPenalty = 0, RightTurnPenalty = 250

Simulation-2 leftTurnPenalty = 0, RightTurnPenalty = 500

Simulation-3 leftTurnPenalty = 0, RightTurnPenalty = 750

Simulation-4 leftTurnPenalty = 0, RightTurnPenalty = 1000

Percentage deviation from the observed values was then considered and the chart prepared (table in Appendix-C)

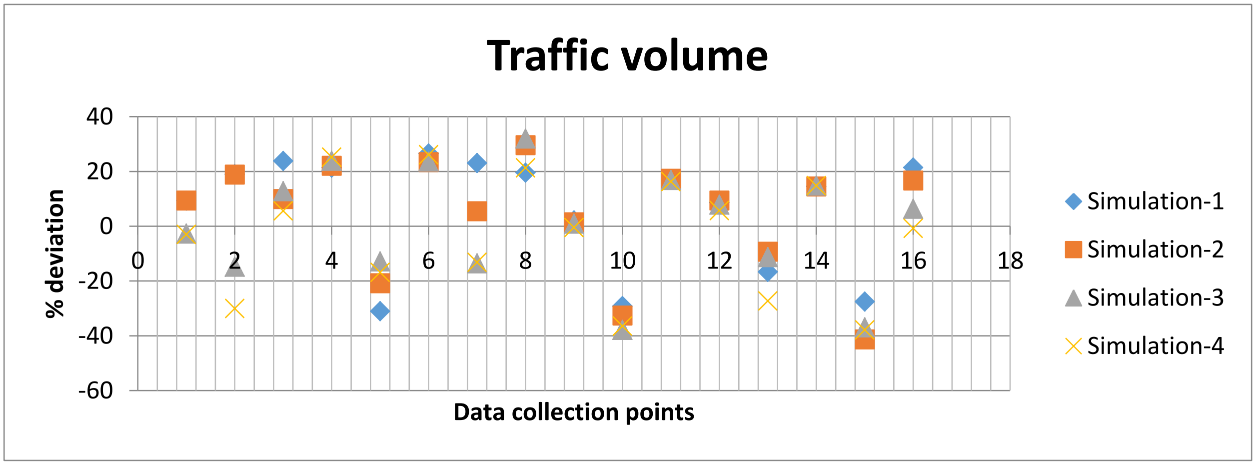


Figure 7.4 Percentage deviations in simulations for different penalty

Then the most suitable guesses for value of right turn penalty for a suitable left turn penalty were tried and for RIGHT_TURN_PENALTY=625, good approximation to the observed data was found, as shown in Fig.7.5

We also observed that no such fixed value of penalties were serving very suitable to give the exact match, which may though be further found by recalibration of other parameters –

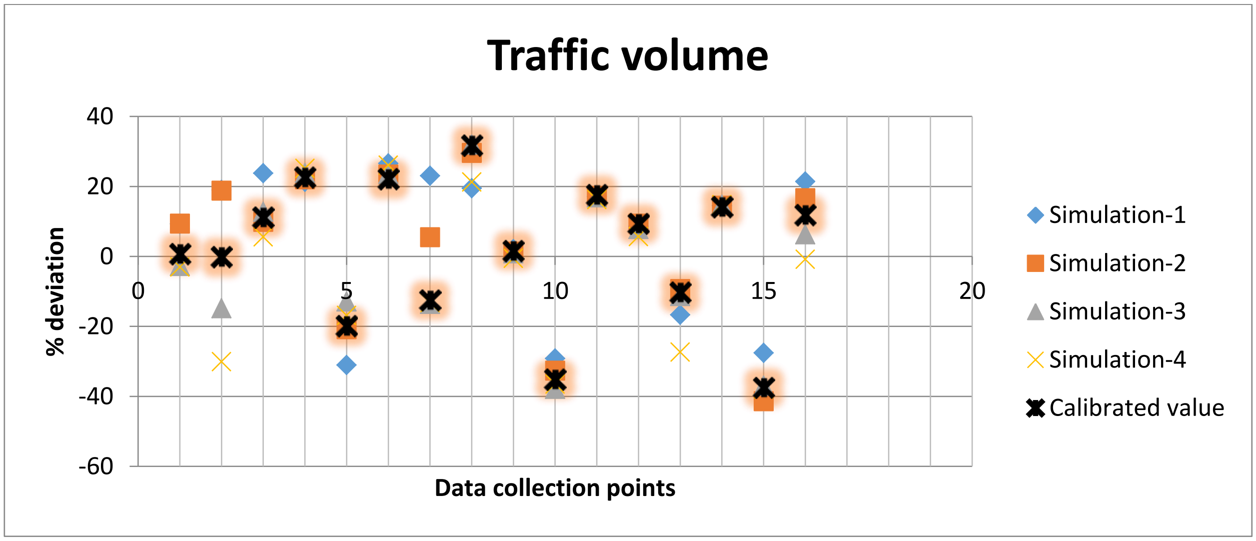


Figure 7.5 Calibrated penalty value deviations

For each of the simulation cases the values of other parameters were kept fixed.

Though the next question might be whether or not other routing and microsimulation parameters effect the simulation output that much? An approximate answer to this goes that as per conceptual logic parameters like Left and Right turn penalties have major say in deciding routes and thus changing the travel pattern of the network. Other parameters like look ahead distance, slow down probability get applied equally on all traveler and have very minimal influence on route choices. Their influence on travel time and travel volume hardly comes into picture if an aggregate hourly analysis is being used to calibrate.

7.4 Convergence: Deciding no. of iterations

As described in the theory, router and microsimulator keep performing iterations until Dynamic user equilibrium is reached. The criteria used in this simulation for convergence to equilibrium was –

"At equilibrium, no user will be able to change its route that will change the travel time for the route by more than 10% of its current travel time in next iteration."

This criteria though a bit relaxed, suits well for the smaller networks, where the actual travel times hold smaller value typically in few minutes. Plan_select program was used to decide on no. of households which did not satisfy the criteria above and only those households were re-routed and entire system re-simulated to find whether the users now satisfy the equilibrium or not.

Based on that criteria it was found that after 15 iterations the no. of households selected for re-routing becomes approx. equal to zero, implying DUE is reached given our criteria.

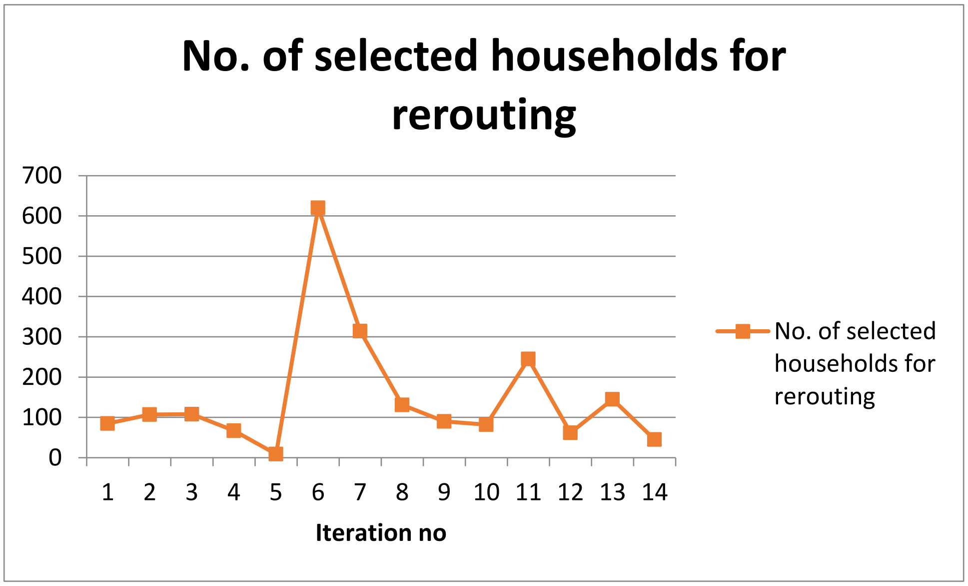


Figure 7.6 Convergence criteria

As observed after 15 iterations of router, only 45 out of 3000 households get selected for re-routing for not satisfying the equilibrium criteria, which is like just 1.5% of the total no. of households. Thus 15 can be assumed a good number of iterations after which the DUE is achieved.

There is a slight random variation of selected households in the start and that is attributed to the behavior that for first 5 iterations PLanSUm uses link travel time function to decide households, which is not an effective way. After 6^{th} iteration, Msim real results provide an updated second to second delay value which thus becoming useful in rerouting and increases the no. of selected travelers.

Chapter -8 Summary and Conclusion

8.1 General

We have observed in the previous sections that TRANSIMS being an open-source software with its applicability spread to larger regions of simulation, has high potential of being developed for solving simulation problems of larger networks and testing for various improvements. To apply it for Indian traffic conditions one needs to consider following adaptations that needs to be made-

a) Introducing new vehicle type with desired vehicular characteristics
b) Introducing the influence of left-side driving
c) Introducing non-uniform mixing of traffic within lanes and involving pedestrian to vehicle interaction by changing the way software deals with vehicular movements

Introducing new vehicle type was already accomplished in TRANSIMS by introducing suitable changes in code (Saxena 2013).Left-side driving incorporation has been dealt in this report(and summarized in sections ahead).

However non-uniform mixing and pedestrian-vehicle interaction are assumed not to be crucial key factors which need not be addressed to a deeper extent. This is because we want to consider behavior of traffic on a larger scale involving million of population for an area closer to 1000sq.km. At such scale effects due to mixing of traffic and pedestrian involvement won't be big, and a usual simulation considering a typical TRANSIMS behavior can bring in to closer accuracy the traffic parameters estimation.

8.2 Left-side driving summary

A left-side driving case is different from a right side driving predominantly at intersections, especially going by the way TRANSIMS implements movements. The travel time calculation done for both the cases would be different and this would mean different cost values would be fed into feedback mechanism of next iteration.

Two methods were proposed out of the study to bring in the required change for left-side driving case –

8.2.1 Changing parameters in router

As analyzed, by keeping suitable values for routing parameters LEFT_TURN_PENALTY and RIGHT_TURN_PENALTY one can bring the same level of disutility that would have occurred if vehicles were moving on left.

A higher value of LEFT_TURN_PENALTY means routes with more left turns would be avoided as they would have higher travel cost. Thus a suitable configuration to achieve left-side driving would be to keep low value of LEFT_TURN_PENALTY and a high value of RIGHT_TURN_PENALTY because a vehicle in a left-side driving scenario would take less time of travel for traversing left turns at intersections, hence a route with a left-turn becomes a preference while routing for next iteration, which is what is achieved by setting low and high values of those parameters.

However this setting of parameters will not change the travel time calculations within iteration and that travel time for vehicles would still be inaccurate. But again what is important here is that, such travel time difference would be hardly 1-2sec per intersection and would be a very small composition of the total travel time of vehicles in entire network.

Thus provided we move towards equilibrium in a correct manner (which we achieve by setting those routing parameters) travel time calculations can be assumed to be accurate and closer to the expected outputs.

8.2.2 Flipping the network

As explained in the sections before, flipping the network by changing its x-coordinates can bring in the desired simulation of left-side driving. The output obtained signified that indeed the modelling is done as per the left-side driving.

Critically this output would yield very accurate results as per left-side driving is concerned because the lane changes prior to intersection and even the signal phase timings, would all be flipped and would be according to the precise left-driving. This would also mean that even the outputs need not be flipped back to analyze correct results.

To be able to visualize such left-side movement in TRANSIMS studio one needs to flip the coordinates back. However the visualization is a tedious task and doesn’t quite show the exact left side movement but at the back-end, processing is done as per the left-side movement. It can thus help in building correct heat plots and identifying where is the concentration of vehicles higher.

8.3 Summary of IIT-Bombay campus simulation

8.3.1 Re-implementation of Track-1 model

To build a correct running simulation model of a region special emphasis needs to be put at

a) Network preparation – for IITB campus it was done by importing shape files of campus and extracting information from their .dbf files. Only node, link and zone files were provided as inputs and rest other elements were left on Transims to build, which were verified at every step

b) Demand input – The real demand of travel activities in the network need to be established. For IIT-Bombay campus trip tables were given as inputs, as part of Track-1 implementation

c) Simulation steps modelling – which steps runs after what gets decided here. In TRANSIMS traditional 4-stage steps had their perspective at a disaggregate level though. Utilties from ConvertNetwork → ConvertTrips → Router → Microsimulation sketch the correct order

d) Output analysis – Text files written as inputs were exported to excel to build useful statistics. ArcGIS shape files and the visualization in Nexta and TransVIS also helped gain the bigger picture of simulation and identify and correct for errors.

Initially developed track-1 implementation of campus model (Neeraj (2013)) was improved at following stages –

i) Improved network configuration – no repeated bi-directional links and added external zone at YP gate

ii) Improved activity locations and signal inputs

iii) Modified for incorrect extrapolation of 1.5hr trip table

iv) Reduced no. of routing errors from 47.6% to 2%, implying simulating close to as many trips as provided in Trip Table

v) Reduced no. of micro-simulation errors from 30% to 1.2%

8.3.2 Flipped network analysis

The flipped network output was compared with the observed data along with the original network values and following conclusions were made –

a) Output of flipped and original network for IIT-Bombay campus, differ only by 1-2%. This is because of absence of signals inside campus, which make left and right side driving very identical
b) The observed data was found in close range ±5 to 25% of the simulated output. Some of the following assumptions would be the cause of differences –
    i) 70% of traffic to external zone goes from main gate and 30% from YP. No such data was made available from previous reports, hence the assumption
    ii) Pedestrian interference with the traffic couldn't be modelled into TRANSIMS accurately even though slightly reduced average speeds were used to compensate for
    iii) Observed data were point volume data, which instead were compared with closest link volume data
    iv) There were still some transims network coding error, with some minor links not being included as part of simulation
c) Both traffic volume and traffic speed data showed better correlation. Traffic speed was far better identical for the reasons that average speed on the links were averaged for all vehicles in a 1-hr period, while volume counts were exact counts of vehicles

8.3.3 Calibration of routing parameters

As observed before, a suitable value of routing parameters was chosen to give least error wrt observed values. Following conclusion can be made regarding the output found –

a) Changes in simulation values (especially traffic volume) for different simulation test runs were radically different following no pattern, implying that just the exact suitable values of penalties can't make a simulation run identical to observed values, but error can be reduced by suitable choice of parameters
b) Since only 5 data points were used to extrapolate for calibrated values, it may be an accurate estimate, but a fairly closer estimate, given hours of analysis work is required to build one data point

8.4 Conclusion on objectives of report

This section analyzes to what extent have the objectives of study been addressed. Track-1 modules of TRANSIMS have been studied for the same and detailed analysis of all the influencing parameters was done. The output parameters were further tweaked in to get detailed level of output until to the extent where memory bounced off, demanding for the need of CPU's with higher processing power.

Left-side driving changes were implemented. Though their verification still needs to be done by comparison with standard software implementing left-side driving, conceptually they hold as true and results obtained above verify the concepts.

All the analysis were visualized in TRANS-VIS that comes in tandem with TRANSIMS-Studio. Both offer an interactive platform for knowing whether what is done is being implemented or not.

Next chapter covers in detail the further things that need to be implemented for better representation of Indian traffic conditions in TRANSIMS.

Chapter-9 Future Scope

9.1 General

Considering what has been covered in the understanding and implementation of TRANSIMS software a large number of future options of improvement remain that can help us adapt TRANSIMS to Indian traffic conditions and to predict to a level of accuracy the realistic traffic parameters for accurate decision making in future. With that broad objective in mind following section summarizes the key areas where efforts need to be put.

9.2 Key areas

9.2.1 Refining track-1 model of IIT-Bombay campus

Current model though implements in detail every step and tries to get the overall picture right, following things indeed if implemented can make it more refined –

a) Providing input shape file which will help us reduce no. of nodes and the position of nodes would be at precise intersections in network

b) Identifying exact schedule of bus services and giving correct input instead of assumption that buses run every 10min. Also the excluded and new routes can be included

c) Conducting re-survey and building new model for updated infrastructure and road network, given several new activity points have come up

d) Moving the activity locations to the precise locations of hostel entrance and exit points. Also trying that parking lot goes where it exists instead of being attached with every activity location

e) Attaching a confidence interval to the values of simulation by performing multiple simulation runs with different random seeds

f) Building suitable model addition to the code where pedestrian interaction with the vehicles can be brought in, by adjusting in Transims implementations

g) Building track-2 model and ensuring that results from it match with updated traffic counts. Though track-2 modelling for such small scale doesn't fruit well, yet a trial in understanding the process needs to be made

9.2.2 Moving to TRANSIMS version 5.0 or 6.0

Though our entire analysis was done and restricted to version-4 simulation and study, for which great documentation was available, research works in newer versions of TRANSIMS namely version 5.0 and 6.0, is taking shape up. Some of the improvements in newer version can be listed down as -

a) Easy smoothened process for building network from shape files
b) Better lane-connectivity problem resolution as formatted details gets provided
c) Easy to do parallel processing on multiple CPU systems
d) Less intensive input files and more coagulated input than separate files

And indeed being updated on the latest release of the software adds major credit to one's work unless it's just a test beta version.

9.2.3 Others

Other options available to be worked out in farther future include –

a) Extending the analysis to the Mumbai-network and identifying how the traffic problems plaguing the city can be solved
b) Incorporating changes in cell-size so that realistic condition of movement of two vehicle side by side in the same lane could be made possible
c) Comparison of the processing time and the outputs by other agent-based modelling softwares like MATSIM, ParaMics etc, for a comparative analysis on which software yields us better convergence, realistic outputs and efficient interactivity.

Chapter 10 – References

1) Chiu, Y. C. et al. (2011, June) *Dynamic Traffic Assignment– A Primer*. Unpublished internal document, Transportation Research Board, Washington DC
2) Jeihani M., Ardeshiri A. (2012), TRANSIMS Implementation for a Small Area Network in Maryland and Comparison with Enhanced Four-Step Model, *Transportation Research Board*
3) Lawe, S., J. Lobb, A.W. Sadek, and S. Huang (2009) TRANSIMS Implementation in Chittenden County, Vermont: Development, Calibration, and Preliminary Sensitivity Analysis. Presented at the 88thAnnual Meeting of the *Transportation Research Board*, Washington, D.C
4) Lieberman, E., Rathi A.,(2001) Chapter 10 Traffic Simulation, Revised Traffic Flow Theory Monograph: [TRB/FHWA version]
5) Saxena N. (2013), "*Developing Simulation model for Indian Road network using TRANSIMS*", M.Tech.Thesis, Indian Institute of Technology, Powai.
6) .Ullah M. S. et al. (2011), Travel Demand Modeling for a Small Sized MPO using TRANSIMS, *Transportation Research Board*, No. 1183, Vol 11
7) Wolshon, B. and B. McArdle (2009) "Temporospatial Analysis of Hurricane Katrina Regional Evacuation Traffic Patterns." *Journal of Intrastructure Systems*, 15(1): p. 8
8) Shan Huang , Liya Guo , Yan Yang , Irene Casas & Adel W.Sadek (2012), "*Dynamic demand estimation and microscopic traffic simulation of a university campus transportation network*", Transportation Planning and Technology, 35:4, 449-467, DOI
9) Rilett, L., Kumar, A. and Doddi, S.(2003), "EL-Paso Transims Case study", FHWA/TX-03/0-2107-1 technical report by Texas Department of Transportation
10) Dowling, R., Holland, J. and Huang, A.(2002) "Guidelines for Applying Traffic Microsimulation Modeling Software", California Department of Transportation, report by Dowling Associates

11) Rilett, L. and Kim, K.O.,(2001), 'A comparison of the TRANSIMS and CORSIM traffic signal simulation modules' , Transportation Research Board, Annual Meeting(2001)

12) Koohbanani, M.J.(2001), 'Enhancements to Transportation Analysis and Simulation System(TRANSIMS), PhD Dissertation, Virginia Polytechnic Institute and State university

13) TRANSIMS Documentation Index, at code Google - <http://code.google.com/p/transims/wiki/DocumentationIndex> as accessed on May 4, 2014

14) Nagel, K., Stretz, Paula., Pieck, M., Leckey, S., and Barrett, C.L. (2008) "TRANSIMS traffic flow characteristics", Pre-print

APPENDICES

Appendix-A Common Errors in TRANSIMS Simulation

To be able to run an error free simulation is considered one of the most crucial measure for ensuring your simulation did go correct. In TRANSIMS while executing every utility we encounter several errors and warning messages, and often being unsure about what causes them we tend to skip them and end up getting incorrect simulation results.

The ‘how-to’ provided under TRANSIMS manual serves as the base for one to understand what possibly is going wrong in simulation set-up. Some of the errors were already discussed in Appendix “TRANSIMS TroubleShooting” of report by Saxena(2012). However some other and crucial ones are given below –

a) All crucial nodes got deleted after running TransimsNet

The answer to this definitely lies in the fact how TRANSIMS views links and nodes in a network. It has difficulty simulating shorter links thus TransimsNet utility combines such smaller links, keeping however the details of length and curvature intact, but in process deleting the nodes.

A good solution would be to make nodes only at predominant intersection or where no. of lanes change. For change of direction of link shape files should be used instead of nodes at every step of curvature.

A turn-around solution would be to keep all nodes under ‘Keep_Nodes_List.txt’ file.

b) Lane connectivity between Link A and D doesn’t exist

It’s a fairly obvious error which points one to exact link where lane connectivity couldn’t get established by TransimsNet. If this is a predominant error with several records flouting the rules, then a thorough check should be made at every intersection in terms of connection angle. Allotting value of parameter MAX_CONNECTION_ANGLE as 180 solves all such problem but should be carefully used as it adds extra connectivity not there in original network. Zone type and link access also play a role in establishing link connectivity. Special consideration should be given in terms of providing lanes in other direction of link, if a separate link with reversed Anode and BNode is not being given. Turn_Prohibition table should also be looked at

c) Park&Ride Node 383 did not have Auto Access

Its probably the error of access restriction with the links. This refers to USE code of the coded network (note imp., not the original network as USE codes go modified after running TransimsNet). Since it relates to park and ride trips which were not part of campus modelling, these errors couldn't get sorted, yet a discussion can be taken place on users-forum.

d) Percent of Total Trips with Problems = x%

One needs to ensure that this report given in Router.prn file should not have value of 'x' more than 5. Typically several of the trips go un-routed because of specific errors like Access_Restriction_error, Vehicle_ID_error, Path building errors, Zero node path problem etc. No specific solution exists in such cases. The network needs to be revisited to ensure that desired coding has happened. Zero-node list errors are not that predominant as they happen for extremely short trips and can be ignored unless that constitutes major proportion. Tips on dealing with individual error can be found in documentation or discussion on forum.

e) Warning: Connections between Links 58 3-3 and 328 1-1 are Not Possible

It's another of a lane connectivity error generated by Microsimulator. Reviewing the lane connections at specified links and lane-ranges helps sort the problem. Special care should be taken to re-execute the utilities if a change in lane-connectivity gets made.

f) Error : No household list records detected

It's a typical error which should rather be taken as achievement than error, and gets generated when PlanSelect is not able to select any household for re-routing, implying that User equilibrium has reached. However other reasons to it must also be looked at like suppose your iteration's linkDelay file had records updated every 1-hour in which we don't have micro-level details of individuals, none of the households might get selected for next iterations.

g) Number of Travelers with Problems = 21215 (43.5%)

It's a microsimulator error signifying the no. of travelers that developed problems while running simulation and couldn't complete their destined plans. This is very crucial as value of this more than 2-3% can cause incorrect travel statistics to be predicted. To sort different types of problems that causes these MSimProblems file should be reviewed and documentation read. Viewing the links causing problems using ArcProblem.exe utility or by joining the table of problem to link layer file, can be possible ways of easy identification.

h) ZeroDivisionError : float division

One of error observed while loading the settings file in TransVIS, it is usually caused by lack of data present in the arcSnapshot. One should be precisely sure that both index file and compressed snapshot file was written out of last utility of Transims studio, and then that the settings control file refer to the respective files correctly.

i) Error converting xxx attribute in yyy file

Most common error telling that there is problem in definition of title of ‘xxx’ attribute in that file. This result from incorrect definition in .def file attached with every input file. A work around can be to delete the .def files and let Transims generate the .def file on its own from our records.

j) No module named transimsRTE found

Its an error when you don’t install python suitably with installation of transims Studio. A work-around to this is definitely reinstallation or transferring the TransimsRTE module from some other working machine to syslib directory of python installed location.

Others of the most common errors have been discussed extensively on the users-forum of TRANSIMS. Typically the errors result only from incorrect network coding or not feeding in values in utilities with proper thought and reasons as specific to the local network. To be specific for to not get errors or warnings during simulation extensive care should be taken before executing every utility after having read the documentation thoroughly.

Appendix-B Data-Tables for the charts used

For figure7.2, we have following data –

Table B.1 Observed vs Simulated bi-directional traffic volumes

Point	Link in transims	Observed Volume	Original network volume	Flipped network volume	original deviation	flipped deviation
1	303	380	344.6667	345	9.298246	9.210526
2	109+108	223	174.333	180	21.82377	19.28251
3	362	185	137.3333	136	25.76577	26.48649
4	329	225	175.6667	174	21.92593	22.66667
5	10+11	101	129	123	-27.7228	-21.7822
6	136	212	159	183.3333	25	13.52201
7	65	91	75.66667	83.66667	16.84982	8.058608
8	121	321	256.3333	252.3333	20.14538	21.39148
9	337	400	385	382.6667	3.75	4.333333
10	143	96	127.3333	148	-32.6389	-54.1667
11	325	387	316.6667	311.3333	18.17399	19.55211
12	6	170	148.3333	148.6863	12.7451	9.210526
13	2	354	405	405.3333	-14.4068	-14.5009
14	332	117	99.66667	99	14.81481	15.38462
15	40	316	404	389	-27.8481	-23.1013
16	353	42	34.66667	35.66667	17.46032	15.07937

For Figure 7.3m we have following data –

Table B.2 Observed vs Simulated travel speed

data collection point	Link	Observed speed	original network	flipped network	original error%	flipped erros
1	41	29.71	24.84	24.84	16.39179	16.39179
2	40	27.38	24.3	24.12	11.24909	11.9065
3	326	29.17	26.64	26.46	8.673294	9.290367
4	328	30.26	27.18	27	10.17845	10.7733
5	332	25.21	27.54	27.9	-9.24236	-10.6704
6	143	27.15	25.02	25.56	7.845304	5.856354
7	64	28.81	24.66	24.66	14.40472	14.40472

For figure 7.4 and 7.5, following tables were used –

Table B.3 Observed vs simulated volumes in different experiments

Point	Link in transims	Observed Volume	Simulation-1 volume	Simulation-2 volume	Simulation-3 Volume	Simulaton-4 volume	Calibrated penalty volume
1	303	380	344.6667	344.3333	390.3333	391.3333	377
2	109+108	223	180.667	181	256	290	223.2
3	362	185	141	166.6667	161.3333	174.6667	164.33333
4	329	225	177	175.3333	171.3333	168.3333	174
5	10+11	101	132.33	122	114	118	121
6	136	212	155.6667	162.3333	162	156.6667	165
7	65	91	70	86	103.3333	103	102.33333
8	121	321	258.3333	226	218.3333	252.6667	219
9	337	400	391.6667	394.3333	396.3333	402	393.33333
10	143	96	124	127.3333	132.3333	131	129.66667
11	325	387	320.6667	320	321.6667	324	318.66667
12	6	170	155.3333	115.4314	156.6667	160.3333	154
13	2	354	413	387	394	450.6667	390.33333
14	332	117	100	100	99.66667	99.66667	100.33333
15	40	316	403	446.6667	432.6667	435.6667	434.33333
16	353	42	33	35	39.33333	42.33333	37

Table B.4 % deviation

	% errors				
Link in transims	simulation-1	simulation-2	simulation-3	simulation-4	calibrated parameter
303	9.298245614	9.385964912	-2.719298246	-2.98245614	0.789473684
109+108	18.98340807	18.83408072	-14.79820628	-30.04484305	-0.089686099
362	23.78378378	9.90990991	12.79279279	5.585585586	11.17117117
329	21.33333333	22.07407407	23.85185185	25.18518519	22.66666667
10+11	-31.01980198	-20.79207921	-12.87128713	-16.83168317	-19.8019802
136	26.57232704	23.42767296	23.58490566	26.10062893	22.16981132

65	23.07692308	5.494505495	-13.55311355	-13.18681319	-12.45421245
121	19.52232606	29.59501558	31.98338525	21.28764278	31.77570093
337	2.083333333	1.416666667	0.916666667	-0.5	1.666666667
143	-29.16666667	-32.63888889	-37.84722222	-36.45833333	-35.06944444
325	17.14039621	17.3126615	16.88199828	16.27906977	17.65719208
6	8.62745098	9.385964912	7.843137255	5.68627451	9.411764706
2	-16.66666667	-9.322033898	-11.29943503	-27.30696798	-10.26365348
332	14.52991453	14.52991453	14.81481481	14.81481481	14.24501425
40	-27.53164557	-41.35021097	-36.91983122	-37.86919831	-37.44725738
353	21.42857143	16.66666667	6.349206349	-0.793650794	11.9047619
Mean error	6.374702037	6.374702037	0.563147828	-3.189671525	1.770749332

As observed, mean error for the calibrated parameter is also close to 0, i.e. 1.77% of overall traffic volume.

For Fig.7.6 following data was used –

Table B.5 Households selected for re-routing

Iterationno	No. of selected households for rerouting
2	85
3	107
4	108
5	67
6	9
7	620
8	314
9	131
10	90
11	82
12	245
13	62
14	145
15	45

Specimen `C' – Declaration

I declare that this written submission represents my ideas in my own words and where others' ideas or words have been included, I have adequately cited and referenced the original sources. I also declare that I have adhered to all principles of academic honesty and integrity and have not misrepresented or fabricated or falsified any idea/data/fact/source in my submission. I understand that any violation of the above will be cause for disciplinary action by the Institute and can also evoke penal action from the sources which have thus not been properly cited or from whom proper permission has not been taken when needed.

____

(Signature)

____

(Name of the student)

____

(Roll No.)

Date: __________

Thank You ☺