INTELLIGENT TRANSPORTATION SIMULATION FOR SUPPORTING SMART CITY FOOD DISTRIBUTION

By

HONG TAT HENG

A REPORT

SUBMITTED TO

Universiti Tunku Abdul Rahman

in partial fulfillment of the requirements

for the degree of

BACHELOR OF INFORMATION TECHNOLOGY (HONS)

COMMUNICATIONS AND NETWORKING

Faculty of Information and Communication Technology

(Perak Campus)

MAY 2016

UNIVERSITI TUNKU ABDUL RAHMAN

REPORT STATUS DECLARATION FORM

Title: INTELLIGENT TRANSPORTATION SIMULATION FOR SUPPORTING SMART CITY FOOD DISTRIBUTION

Academic Session: MAY 2016

I HONG TAT HENG declare that I allow this Final Year Project Report to be kept in

Universiti Tunku Abdul Rahman Library subject to the regulations as follows:

1. The dissertation is a property of the Library.
2. The Library is allowed to make copies of this dissertation for academic purposes.

Verified by,

_________________________ _________________________

(Author's signature) (Supervisor's signature)

Address:

N,85D, Jalan Melayu,

32400 Ayer Tawar,

Perak.

____

Supervisor's name

Date: 26 AUGUST 2016

Date: ____________________

INTELLIGENT TRANSPORTATION SIMULATION FOR SUPPORTING SMART CITY FOOD DISTRIBUTION

By

HONG TAT HENG

A REPORT

SUBMITTED TO

Universiti Tunku Abdul Rahman

in partial fulfillment of the requirements

for the degree of

BACHELOR OF INFORMATION TECHNOLOGY (HONS)

COMMUNICATIONS AND NETWORKING

Faculty of Information and Communication Technology

(Perak Campus)

MAY 2016

DECLARATION OF ORIGINALITY

I declare that this report entitled **“INTELLIGENT TRANSPORTATION SIMULATION FOR SUPPORTING SMART CITY FOOD DISTRIBUTION”** is my own work except as cited in the references. The report has not been accepted for any degree and is not being submitted concurrently in candidature for any degree or other award.

Signature : ____________________________

Name : ____________________________

Date : ____________________________

ACKNOWLEDGEMENTS

I would like to express my sincere thanks and appreciation to my supervisor, Dr. Goh Hock Guan for giving me a great opportunity to engage in the traffic simulation project. Much appreciated for all the guides and advices throughout this project

Last but not least, I must say thanks to my parents and my family for their love, support and continuous encouragement throughout the course.

ABSTRACT

The project title is Intelligent Transportation Simulation for Supporting Smart City Food Distribution. The scope of the project is to simulate part of the smart city's traffic by using Simulation of Urban Mobility (SUMO).

Food distribution process is the main concern in the city especially due to the unpredictable traffic in city which can greatly slow down the distribution process. Hence, it is necessary to come out with intelligent transportation simulation in order to simulate the traffic situation in the city and then simulate the trucks distribution process.

Through the simulation, it is necessary to find out which are the best routing algorithms for the trucks to reach their destination, and then evaluate the routing choices based on the fuel consumption, distribution time, and also the carbon emission. Besides that, it is necessary to find out that by using SUMO, should the features for example, the ability of the trucks to coordinate themselves and reroute automatically when the planned routes is stuck in heavy traffic congestions, these two features would be the greatest challenges to face in this project.

In a nutshell, by simulating and applying the scenarios as mentioned, in the end of the project, we are able to calculate the best routes for the food trucks to distribute their logistics, and then when traffic congestions happen, then they are able to reroute themselves to ensure they have the best timing to distribute the logistics to the shops.

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

LIST OF ABBREVIATIONS

GHG	Greenhouse Gases
MT	Map and Transfer
OSM	Open Street Map
SUMO	Simulation of Urban MObility
VRP	Vehicle Routing Problem
SSM	Soft System Methodology
Trass	TraCI as a Service

CHAPTER 1: INTRODUCTION

1.1 Motivation and Problem Statement

"For most of human history… perishable foods were by definition local. They travelled far only if… they could be kept alive and breathing." (Susan Freidberg, 2009, p.408).

The foods grown within the city are usually perishable or semi-perishable, for example fruits and vegetable like broccoli, lettuce, carrots, potatoes, etc. These foods need to be kept in appropriate temperature after harvesting to ensure the freshness and the nutrient contents do not deplete rapidly. *MIT's Mission 2014: Feeding the World (2014)* mentioned the extent of waste that occurs post-harvest and during transport whereby most of the produce is very perishable and is susceptible to bacteria, insects, and fungus that rot the food and contaminate it with disease, rendering the food inedible. They estimated that 25%-50% of all food produced is wasted and hence, causing the food shortage as well as the income of producers are suffering losses.

Nevertheless, issues araised when these foods need to be distributed from the producers to the retailers whereby the foods are temporarily stored in large refrigerated warehouses and transported to supermarkets, restaurants and retailers. According to Clement et.al (2009), approximately 40% of the vegetables never make it to the supermarket shelves due to the damage during the transit. It is a great challenge for food distributors to transport food from field (producers or processors) to retail (supermarkets, restaurants and etc.) as they need to make sure that the foods are well kept and fresh during the transportation period. In the past research by *Clement et.al (2009) and Wayne et.al(2012)*, they listed out few factors affecting produce quality during transport. These factors are initial quality, temperature, humidity and water loss, atmospheric composition, mixed loads and physical injury. Therefore, the time to distribute the foods is very essence as it determines the freshness of the foods when they are distributed to the retails.

Since we know that temperature and humidity are the main influence to the quality and freshness of foods, according to *Maurizio and Olivio (2011)*, there are few issues in transporting the foods within the city which included the norm of traffic congestions in the city as well as lack of the parking space to unload the foods from the trucks. When the trucks that are responsible to distribute the foods to designated destinations are besieged in the heavy traffic, the foods inside are suffering quality degradation as well due to the increasing heat and water losing with increasing time. Besides that, the stop and move motion of the trucks during the traffic congestion will also cause physical damage to the foods due to the vibration and compression.

On top of that, besides the issues of quality of food during transportation process, people are also concerning the pollution causing by the trucks. NRDC's Food Miles (2007) and Johns Hopkins Center's Food Distribution And Transport raised out the energy and climate consequences of food transport whereby the fossil fuels used by the transportation vehicles will emit greenhouse gases(GHGs) that contribute to climate change.

The statistics below showed the results of transportation related impacts of importing agricultural products into the state's ports (Los Angeles, Long Beach, Oakland, and Otay Mesa on the Mexican border) by NYDC:

"Almost 250,000 tons of global warming gases released were attributable to imports of food products—the equivalent amount of pollution produced by more than 40,000 vehicles on the road or nearly two power plants.

More than 6,000 tons of smog-forming nitrogen oxides were released into the air—the equivalent of almost 1.5 million vehicles or 263 power plants!

300 tons of sooty particulate matter were released into the air—the equivalent of more than 1.2 million cars or 53 power plants. " *(NYDC, 2007)*

As a whole, the longer the time the food distributed to the retailers, the more negative impacts they are to the quality of food and the air pollution. In the end, the food distributors and the citizens are the one who suffered. This is because the foods distributed which are already rotten or damaged during the long hours of transportation due to the unpredictable traffic cannot be sold to their customers, therefore, the food distributors have to bear the profit lost. At the same time, when there are insufficient supply of food to the retailers but the demand is high, then there will be an increase of food price which the consumers have to bear as well.

Therefore, there’s a need to solve the issue whereby the unpredictable traffic within the city by having an intelligent transportation simulation in order to support the food distribution within the city whereby drivers are allowed to change to a better path based on the simulation in case any sudden situations happened in the traffic. Hence, to shorten the food distribution from the producers to the retailers.

1.2 Project Objectives

The main goals of this project are as follow:

- to simulate the traffic situations in cities with food trucks transportation loading and unloading logistics capability by using a traffic simulator
- to simulate the transportation process of the food trucks
- to suggest the best routing algorithm for the food trucks to distribute logistics based on the simulation
- to evaluate the performance of the routing algorithms with following criteria:
    - petrol consumption
    - Carbon emission
    - the time needed to reach the destination to unload the logistics

1.3 Project Scope

In this project, only the part of smart city’s traffic whereby the location which has high probability to cause the traffic congestions is simulated because modelling a whole smart city is impossible due to its complicated behaviour.

Taking Kuala Lumpur which is a big city in Malaysia as example, due to the limitations of SUMO which supports a maximum of 10,000 edges in the network, Kuala Lumpur map couldn’t be implemented. Figure below shows the error when trying to import Kuala Lumpur map in SUMO.

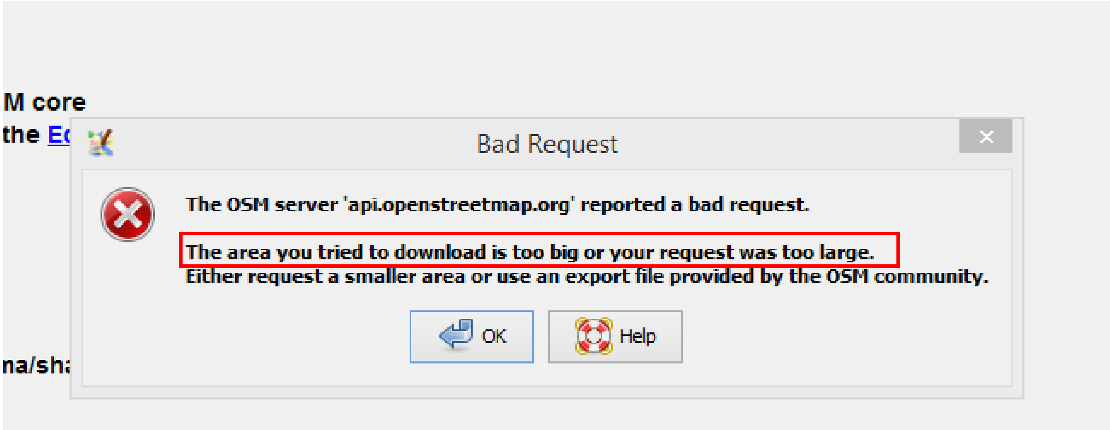


Figure 1.3.1: Bad Request when trying to import Kuala Lumpur city map

Hence, Subang Jaya city which is located in Selangor is implemented in this project. Worldatlas (2015) has stated that the population of Subang Jaya city is 708,296 and it is the third biggest city in Selangor.

There are few cases which can be simulated in this project, example cases like having traffic jams on the road. Besides that, the simulation of the estimated time that the food trucks required to deliver foods will be conducted as well.

All those simulations will be done by using a traffic simulator known as Simulation of Urban Mobility (SUMO). The reason why SUMO is adopted is because it is an open source tool which own algorithm can be implemented. According to *Michal Behrisch et.al (2011)*, there are some open source traffic simulations available, but most of them have been implemented within a student thesis and got unsupported afterwards. In addition, *David Smith et.al (2014)* stated that this simulator allowed us to generate traffic demand on a small grid network which enabling us to see what was happening during runtime because OpenStreetMaps (OSM) are often too complex to see exactly what is happening.

On top of that, DLR's documentation has listed out the features of SUMO which stated as below:

- Includes all applications needed to prepare and perform a traffic simulation
  (network and routes import, DUA, simulation)
- Simulation
- Space-continuous and time-discrete vehicle movement

- Different vehicle types
- Multi-lane streets with lane changing
- Different right-of-way rules, traffic lights
- A fast openGL graphical user interface
- Manages networks with several 10.000 edges (streets)
- Fast execution speed (up to 100.000 vehicle updates/s on a 1GHz machine)
- Interoperability with other application at run-time
- Network-wide, edge-based, vehicle-based, and detector-based outputs
- Supports person-based inter-modal trips
- Network Import
- Imports VISUM, Vissim, Shapefiles, OSM, RoboCup, MATsim, openDRIVE, and XML Descriptions
- Missing values are determined via heuristics
- Routing
- Microscopic routes - each vehicle has an own one
- Different Dynamic User Assignment algorithms
- High portability
- Only standard C++ and portable libraries are used
- Packages for Windows main Linux distributions exist
- High interoperability through usage of XML-data only
- Open source (GPL)

1.4 Main Contributions from the Project

Through this project, it is expected to serve as a platform to help scientific research by contributing vehicle mobility data. Sumo can be used as a developing tool for those researches who need vehicle mobility data with minimal knowledge of the underlying map convertors and microscopic vehicle simulators.

Besides that, this project can become a potential study in changing the behaviour of the traffic for example when urban planners need to measure the effect of vehicle mobility in urban area in any change of city road network, performance of traffic light in order to smoothen the flow in traffic, provide traffic surveillance and also to provide further study of the vehicle route choices.

1.5 Organisation of the Report

This report includes a total of seven chapters. Chapter 1, which is the Introduction of this entire project discusses the motivation and problem statement, project objectives and scope as well as the expected contribution of the project.

In Chapter 2 which is the Literature Review, includes the review of the existing applications and the summary of them so that readers will have a clear understanding the advantages and also the disadvantages of the existing applications.

System Methodology which is discussed in Chapter 3 included system development models, the technologies involved in the project, the functional requirement, expected system testing and performance, challenges, the project milestone as well as the estimated cost for the entire project.

In Chapter 4, the simulation process which discusses how's the map is imported, and the generation of trips and route files so as to run the simulations. The logic behind where the implementation of Backup Trucks is described as well following by the simulation environment setup.

Whereas in Chapter 5, the implementation and configurations of the simulation will be described together with the implementation of logic into coding form using Java Eclipse. Besides that, the simulation operations of the trucks are illustrated through the screenshots.

Chapter 6 includes the simulation results and evaluation of the project. Project challenges and also the objectives evaluation of the projects will be discussed as well.

Chapter 7 is the conclusion of the whole project following by the recommendation throughout the entire project.

CHAPTER 2: LITERATURE REVIEW

2.1 Review of the Technologies

There are three traffic simulators which will be reviewed, which are:

a. Simulation of Urban Mobility (SUMO) (Sumo 0.27 2016)

- SUMO is a microscopic road traffic simulation package which is of open source and of highly portable

b. Treiber's Microsimulation of Road Traffic (Microsimulation of Traffic Flow:html 5 version 2016)

- This is an open source traffic simulator created by Triber for the purpose of research work in traffic dynamics and traffic modelling

c. Aimsun (Aimsum 8.1.3 2016)

- there are three types of transport models included in Aimsun simulator which are static traffic assignment tools, mesoscopic and microsimualtor

These three simulators will be reviewed based on their software category, User Documentation and Graphical User Interface (GUI), Network traffic generators and vehicle patterns/types and large traffic networks simulation ability and quality of graphical representation.

2.1.1 Software Category

Before choosing the simulators to be used, it is necessary to know that they are for commercial use or for open source. Open source is most welcomed because it provides the right for users to use, study and modify without any restriction.

Among the three traffic simulators mentioned, both SUMO and Treiber's Microsimulation of Road Traffic are free to use, whereas Aimsun is commercial software provided with 30 days trial. Due to the free use and open source of SUMO and Treiber's Microsimulation of Road Traffic, their source codes are able to be downloaded for free and modified.

2.1.2 User Documentation and Graphical User Interface (GUI)

Traffic simulators are basically quite complex to use and also, providing a good user interface will be important for the user to get handy on it. Both SUMO and Treiber's Microsimulation of Road Traffic are having user documentations. However, Treiber's Microsimulation of Road Traffic user documentation is not as detailed when comparing to SUMO's user documentation which are divided based on each features accordingly, and at the same time, tutorials with examples are provided as well for better understanding. According to the study by G.Kotusevski et.al (2009), no user manual is provided and after going through the website at https://www.aimsun.com/, the training is only provided when the user buys the license from Aimsun.

In terms of GUI, due to the provided user documentation, both the open source simulators are relatively understandable however again, according to G.Kotusevski et.al (2009), Aimsun GUI is more challenging which is mainly caused by lack of provided user documentation.

2.1.3 Network traffic generators and vehicle patterns/types

In SUMO, it is quite straightforward to generate the traffic on the imported map since the system provided a tool known as DUAROUTER where by it can perform the function as below:

- Building vehicle routes from demand definitions

- Computing routes during a user assignment

- Repairing connectivity problems in existing route files

According to G.Kotusevski et.al (2009), in Treiber's Microsimulator the number of vehicles emitted per hour from a certain intersection as well as the ratio between cars and trucks in certain scenarios can be defined by the user based on statistical distribution of vehicles. Whereas the traffic network in Aimsun can be manually drew using the graphical network editor.

For the vehicle types, both SUMO and Aimsun are able to simulate various types of vehicles but not the case for Treiber's Microsimulator as it only able to simulate both the cars and trucks.

2.1.4 Large traffic networks simulation ability and quality of graphical representation

All of the 3 simulators are able to simulate the large networks, for example a large city. In SUMO, it stated that it is able to simulate up to 10,000 of edges.

Whereas in terms of graphical representation during simulation, it relatively important so as to allow users to observe the traffic situation clearly and spot any important events happening for example, misbehavior of traffic flow or traffic pattern. Both SUMO and Treiber's Microsimulator only support 2-dimensional view whereas Aimsun is able to provide a 3-dimensional view.

2.1.5 Summary of the Technological Review

Features	**SUMO**	**Treiber's Microsimulator**	**Aimsun**
Software Category	Open Source	Open Source	Commercial with 30 days of trials
User Documentation	Provided in full with updates from time to time	Limited	Not provided for trials version
(GUI)	Easy to understand	Easy to understand	Complex due to the lack of user documentation
Network Traffic Generators	- Straight forward with DUAROUTER - using XML files	Using statistical distribution of vehicles	can be manually drew using the graphical network editor
Large Traffic Networks Simulation Ability	Yes	Yes	Yes
Vehicle Types Simulation	Supports all	Only car and truck	Supports all
Quality of Graphical Representation	2-Dimensional	2-Dimensional	3-Dimensional

Table 2.1.5.1: Summary of the Technological Review

Based on the three traffic simulators, SUMO is the ideal simulator to be used in this project. The main reason is because it is open source and provided with lots of user documentation which updated from time to time. Treiber's Microsimualtor is open source as well, but the user documentation is limited and the vehicle support types are only car and truck which is not enough in this project because in a city, there will not only be cars and trucks but public transports are existing too.

Even though Aimsun is more complete and easier in terms of network traffic generation, the downhill is that it is only a commercial use with only 30days trials which is not enough for my entire project. This is the reason why SUMO which is totally free to use and with customizable source code is chosen for this project.

2.2 Review of the Existing Systems/Applications

2.2.1 Daily Truck Routing and Scheduling software (VRASS)

In another approach by Kuladej and Peerayuth, they attempted to develop a daily truck routing and scheduling software (VRASS) which will improve the efficiency of the perishable food of food company in Bangkok whereby previously, the company scheduled and routed the trucks manually using the experience of the staff person. Hence, this implies that the process is not operating at optimal level and is not efficiency.

The software is developed based on the algorithm by Tillman and Cochran (1968) which extended from Clarke and Wright Algorith (1964). The tasks of this software is to develop task sequencing and then trucks routing to deliver perishable foods in order to minimize the total distribution distance.

Through this software, the total distribution distance is improved by decreasing the overall distance. Besides that, this software can generate the schedule in various routes in any tour which is more flexible than the existing schedule by the experience of the company officer. Their VRASS Schedule is able to utilize the resources in a well manner and also, the planning time is also shorten.

However, this software only assumes that the distance and route between each pair of node is symmetric, but it does not include the way that Metropolitan city with unpredictable traffic which cause it not applicable and ineffective in the real time situation when dealing with those situations.

2.2.2 Application of Lean Methodology to the Field of Supply Chain Management

Javier et.al (2013) came out with an application of lean methodology to the field of supply chain management, and in particular, to routing operations so that transportation agility and food freshness can be increased. This methodology is then applied to Tyson, a leading Mexican food processor as they are facing competitive pressure whereby they need to have more efficient way to distribute the perishable foods in timely manner to ensure the freshness of the foods and hence gain the reputation from their customers.

In order to reduce the food distribution time, this methodology contributes with a two-level approach to identify and eliminate specific waste associated with the chain and the transportation of goods to improve its efficiency.

First of all, waste identification for the distribution company need to be carried up and enriched with the use of value stream mapping. There are two levels of waste identification which are at chain level and at each facility or route level. The figure below shows the general structure of transportation waste reduction scheme by Bernavardo et.al (2012):

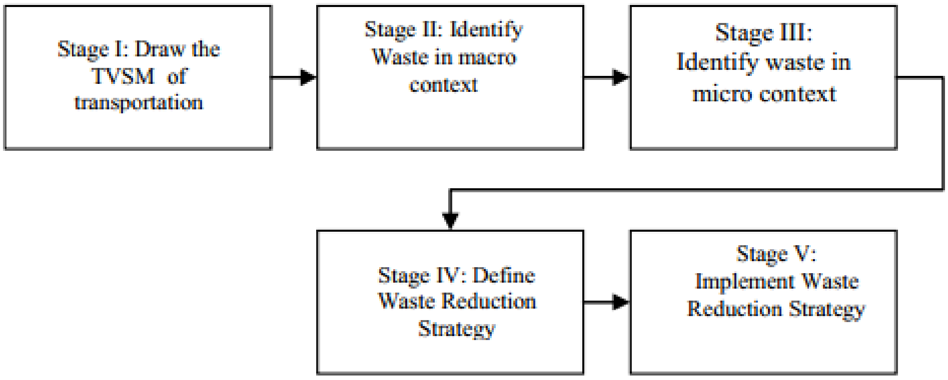


Figure 2.2.2.1 General structure of transportation waste reduction scheme

The next stage is to identify the waste in macro context as well as the micro context.

“The macro context is directed to identify the macro characteristics of the route, namely; Average journey duration, the modified TOVE index and its components, vehicle administrative availability utilization based upon calendar time, availability wastes occurring off the route (such as vehicle non-scheduled time and scheduled maintenance time) and the proportion of internal and external activity time.” Bernavardo et.al (2012). Whereas, in micro level, waste that impact on performance, operating availability and quality factors are taken into account.

Following step is to define the waste elimination strategies at chain and installation levels. Strategies like inventory reduction strategies, facility relocation and transportation mode change could be used to eliminate waste at chain level. Whereas strategies like increasing performance, availability and quality efficiencies are applied on installation levels. The final step is to implement those strategies to the chosen company.

They have applied this methodology to the distribution network of Tyson, a Mexican company leader in the production and distribution of frozen and refrigerated food and there's an important reduction of total cycle time of 46%, a reduction of 35% in the number of routes is achieved. Fill loss is decreased 60% and routing cost is also reduced by 31%. These results have been considered as very promising by the management of the company.

However, in this methodology, even though they are able to identify and eliminate the factors that may cut down the transportation time, they didn't consider the unpredictable traffic that would affect the transportation time. Hence, the effort to reduce the factors that causes time wasted during transportation will become negligible when the food trucks faces heavy congestion on the road which will then degrade the food freshness in the end.

Therefore, it is necessary to provide a route simulation program to deal with the unpredictable traffic that faced so that, with combined waste lean methodology that implemented, the distribution time and cost will greatly improved in smarter way.

2.2.3 Two-stage Solution Procedure for Food Delivery Decisions in Cities

In a study by Wang Zheng et.al (2007), they are trying to analyse the decision making of fresh food transportations issues in cities in conjunction with the transportation systems which are constructed with parallel loops. However, the efficiency of the existing vehicle routing problems (VRP) solution maybe degraded due to the reason whereby parallel loops without traffic lights are the backbones of most of the cities' transportation systems. This has made the task of distributing foods to the retailers in timely and cost-effective manner becomes complicated.

Hence, they came out with a two stage solution approach in order to solve this problem which are “enumeration of vehicle routing schemes” stage and “model construction and solution generation” stage. In the first stage, they used Tree-like Search Graph to enumerate the vehicle routing scheme in order to identify all possible travel schemes for each vehicle type.

In second stage, they constructed a mathematical model to determine the optimal number of vehicles to be used which is to ensure that the number of deliveries completed in one area should be at least equal to the number of customers in the specific area. Then they used a Sweeping Method in order to assign customers for a vehicle. Through these 2 approaches in the second stage, they are able to select the best routing schemes for the vehicles. The figure below shows the Sweeping Method that described.

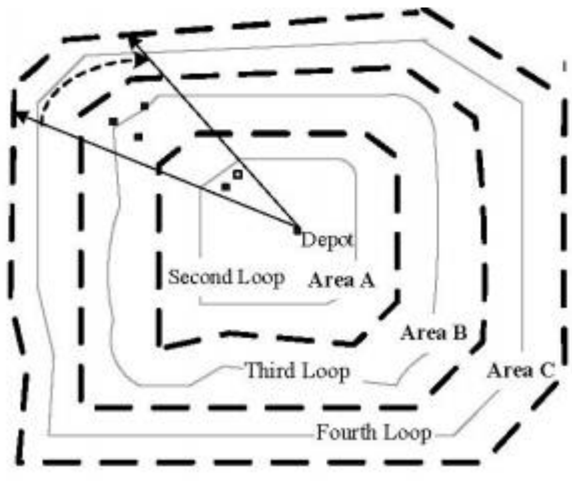


Figure 2.2.3.1: The Sweeping Method for Assigning Customers for a Vehicle

Through this two-stage solution process, the decision making time can be fasten up from few hours in the past to less than an hour after they applied this solution to a grocery company which faced the issue of delivering the foods on time. Next, the scheduling and routing decisions for each day can be made promptly after all the orders are received and the solution can be applied to a wide range of vehicle routing problems with circular transportation networks and fixed delivery time window.

However, in this approach, the efficiency of the process will be decreased with the increasing number of locations in the express loops and the number of vehicle types. Besides that, this solution is limited to the cities with transportation systems which made up of several parallel loops that have no traffic lights and it is not efficient if it applies to cities with lots of traffic lights. Finally, in the situation where there is a sudden incidents happens during the journey, this solution is unable to provide another routing schemes which is effective since the routings are already planned before the foods are distributed. The inflexibility of this approach is the major drawbacks and hence, it is necessary to come out a solution to face the sudden incidents that happened on the road and then provides an alternative routes to the vehicles in order to avoid the congestion and obstacles.

2.2.4 An Actual Urban Traffic Simulative Model (AUTM) for Predicting and Avoiding Traffic Congestion

In the paper by Wenbin Hu et.al (2014), they developed and tested an actual urban traffic simulative model (AUTM) for predicting and avoiding traffic congestion. They are trying to overcome the limitations of the previous work by Biham, Middleton and Levine(BML) which was a two-dimensional cellular automata (CA) model whereby this BML model lacks the function of optimization, and don't have the ability to optimize traffic flow and avoid the traffic congestion.

There are three contributions in AUTM which are the map and transfer (MT) conversion to get the actual urban corresponding cellular space. Next, are the spatial evolution rules and a congestion-avoidance routing algorithm which makes use of the knowledge from global simulative real-time information and allows vehicles to dynamically update their routes towards their destinations.

In the MT conversion method, they performed three steps which are mapping specific urban road networks to corresponding topological graphs by four typical conversion. Next, transferring every links of urban topological graph to the corresponding cellular lattice sequence in their cellular space and finally, setting appropriate value of R for each lattice site to achieve whole equivalent conversion.

Whereas optimizing special evolution rules are applied to vehicles to move on cellular lattice sequence of two-way multilane roads without explicit lane changing rules. In the final steps whereby congestion-avoidance routing algorithm is applied, which is when drivers know the shortest paths to their destinations.

After applying AUTM to six different cities, it can be seen that AUTM has better performance in traffic congestion forecast and simulating the effect of adding overpasses and roadblocks. Their results have shown that the traffic congestion forecasting accuracy is more than 89%. Besides that, the congestion-avoidance routing algorithm is able to improve the traffic flow under discrete local congestion.

Even though this approach is able to predict and avoid traffic congestion effectively, however, in real time environment, there are many unpredictable traffic that could happen and these situations are hardly to be predicted, for example, a road accident, breaking down of traffic lights, etc. Therefore, re-routing the vehicles to avoid these situations on the spot become ineffective since this approach is limited to predict the traffic congestion issue in the cities.

Hence, to improve this approach, features like real-time vehicles re-routing to bypass the blocked road due to an incident can be added to make it more flexible and intelligent.

2.2.5 Summary of the Existing Systems

Literature Review	Advantages	Disadvantages	Comments
1 Daily Truck Routing and Scheduling software (VRASS)	- is able to generate the food distribution schedule in various routes in any tour which is more flexible - utilize the resources in a well manner and also, the planning time is also shorten.	- does not assume the way that Metropolitan city with unpredictable traffic	- Need to consider on the unpredictable traffic in real time situation
2 Application of Lean Methodology to the Field of Supply Chain Management	- able to identify and eliminate the factors that may cut down the food transportation time	- doesn't consider the unpredictable traffic that would affect the transportation time	- Need to consider on the unpredictable traffic in real time situation
3 Two-stage Solution Procedure for Food Delivery Decisions in Cities	- improve the decision making time to decide food distribution routes and schedule from few hours in the past to less than an hour - scheduling and routing decisions for food delivery process each day can be made promptly after all the orders are received	- inflexible to provide another routing schemes which is effective since the routings are already planned before the foods are distributed	- Need to consider on the unpredictable traffic in real time situation

4 An Actual Urban Traffic Simulative Model (AUTM) for Predicting and Avoiding Traffic Congestion	- able to predict and avoid traffic congestion effectively based on the simulation	- ineffective to deal with unpredictable traffic situations on the spot	- Need to consider on the unpredictable traffic in real time situation - Provide vehicles re-routing to bypass the blocked road due to an incident on the spots

Table 2.2.5.1 Summary of the Existing Systems

2.3 Concluding Remark

In conclusions there are some limitations on the approaches mentioned above. When distributing the foods in the city, unpredictable traffic may occur, for examples, heavy jam in certain areas, accidents or road closed for certain events and etc. When these situations occur, those approaches are not applicable and effective as they are pre-scheduled and not flexible to deal with the sudden changes during the journey.

In order to solve the limitations faced, intelligent transportation simulation is necessary to deal with the unpredictable traffic flow during food distribution. This simulation should be able to compute the shortest path to the destination when the pre-selected path is having heavy traffic congestion. Besides that, the trucks that assigned at specific zones are freely to distribute food to another zones which is not under their responsibility when the trucks in charge in that zones found out to be late to reach the destination provided the trucks are near to that locations through the simulation program based on the real time computation.

As a whole, by applying the intelligent transportation simulation, the time to distribute the foods can greatly shortened as it is smart enough to deal with the sudden changes during food distributions and so, the freshness of foods are able to be kept and avoid heavy emission of GHGs during traffic congestion.

CHAPTER 3: SYSTEM METHODOLOGY

3.1 System Development Model

In this project, Soft System Methodology (SSM) will be used in order to carry out this project whereby it is an approach to look inside the problem situations perceived to exist in the real situation. In the case of investigating unstructured problem situation, philosophy and set of techniques in SSM can be applied and it is a way of system requirement analysis instead of system design approach.

According to Susan Gasson(2013), most system requirement analysis techniques are focusing on the way that computer system operates, however, system of work is the focuses of SSM which is the "human activity system" that requires computer system support. Both the information system requirements and information system design are not produced by SSM, instead, set of feasible actions which can be implemented to enhance the problem situation is produced using SSM. The figure below summarize the purpose of Soft System Methodology.

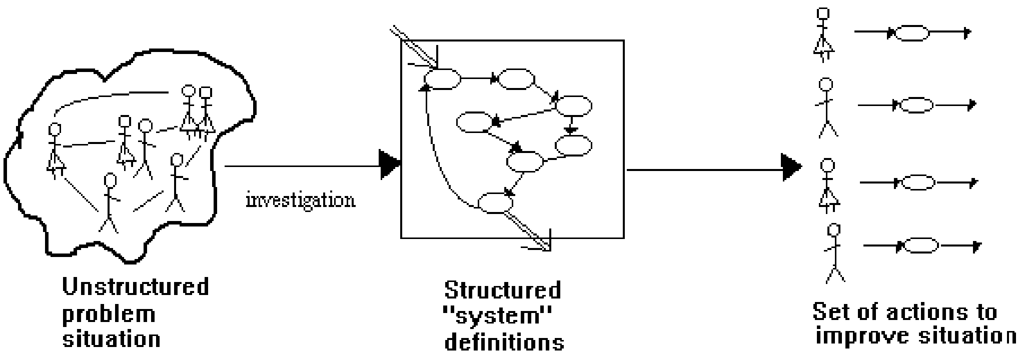


Figure 3.1.1: Summarization of Soft System Methodology

There are a total of 7 stages in SSM which are:

- Investigating the unstructured problem
- Problem situation expression
- Root definitions of relevant systems of purposeful activity
- Conceptual Models building
- Comparing models with real world situations
- Feasible and desirable changes
- Actions to improve the situation

In this project, Soft System Methodology would be the right choice to apply. The real world situation whereby the food quality issue during transportation process is first investigated and then information gathering to the cause of this issue for example the unpredictable traffic situation in the city is then performed in the second stage.

In root definition stage, we can know that root of the problem is that lack of a system to deal with the unpredictable traffic during the transportation process, and hence, simulation of the city with logistics components is needed in the next stage. Through the simulation, different traffic situations are simulated with different routing algorithms, and then the simulation result is then compare with the real world actions. If there's deviation with the real world situations then parameters in the simulation would need to be redefined in order to fulfill the requirement in the real world.

3.2 Technologies Involved

3.2.1 Hardware

- Laptop with a Microsoft Windows based operating system

3.2.2 Software

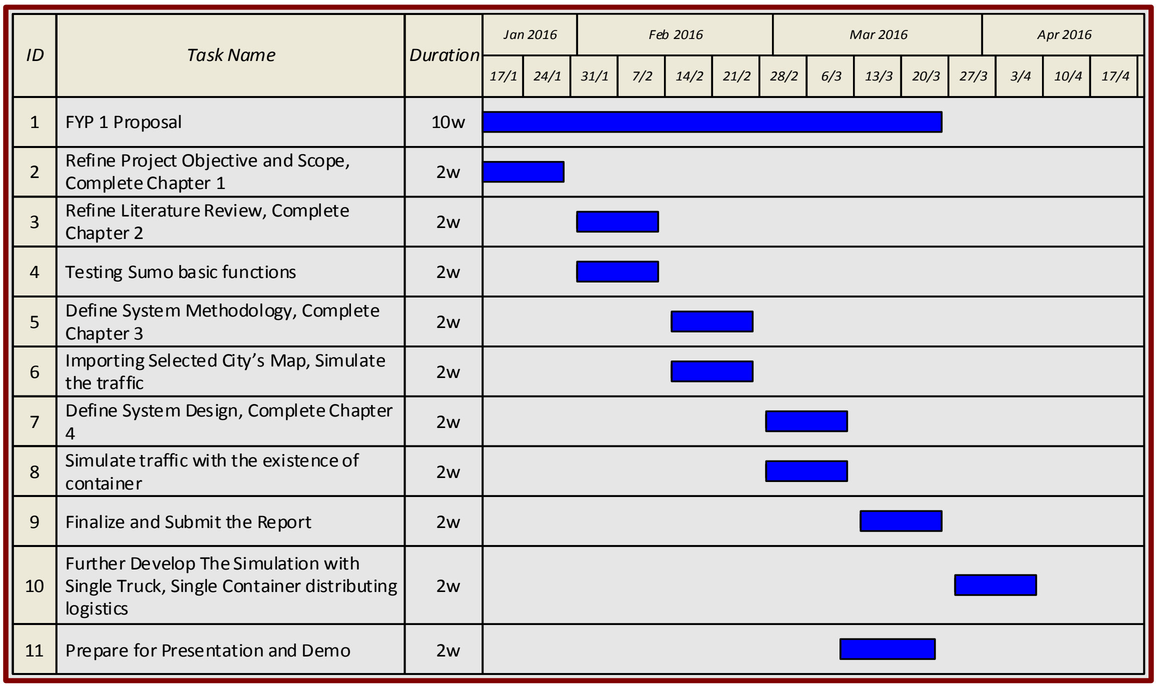
- SUMO "Simulation of Urban Mobility" traffic simulation engine which can downloaded free from http://www.dlr.de/ts/en/desktopdefault.aspx/tabid-9883/16931_read-41000/

- Open Street Map which is a free editable map of the world to be imported into SUMO

- Java Open Street Map is a tool used to import and edit Open Street Map

- TraaS, TraCI as a Service is a web-service adapter for TraCI which allows automatic API generation for multiple languages and it is an essential tool for demand modelling in SUMO

3.3 Project Milestone



ID	Task Name	Duration	Jan 2016		Feb 2016				Mar 2016				Apr 2016			
			17/1	24/1	31/1	7/2	14/2	21/2	28/2	6/3	13/3	20/3	27/3	3/4	10/4	17/4
1	FYP 1 Proposal	10w														
2	Refine Project Objective and Scope, Complete Chapter 1	2w														
3	Refine Literature Review, Complete Chapter 2	2w														
4	Testing Sumo basic functions	2w														
5	Define System Methodology, Complete Chapter 3	2w														
6	Importing Selected City's Map, Simulate the traffic	2w														
7	Define System Design, Complete Chapter 4	2w														
8	Simulate traffic with the existence of container	2w														
9	Finalize and Submit the Report	2w														
10	Further Develop The Simulation with Single Truck, Single Container distributing logistics	2w														
11	Prepare for Presentation and Demo	2w														

Table 3.3.1: Gantt Chart for FYP 1

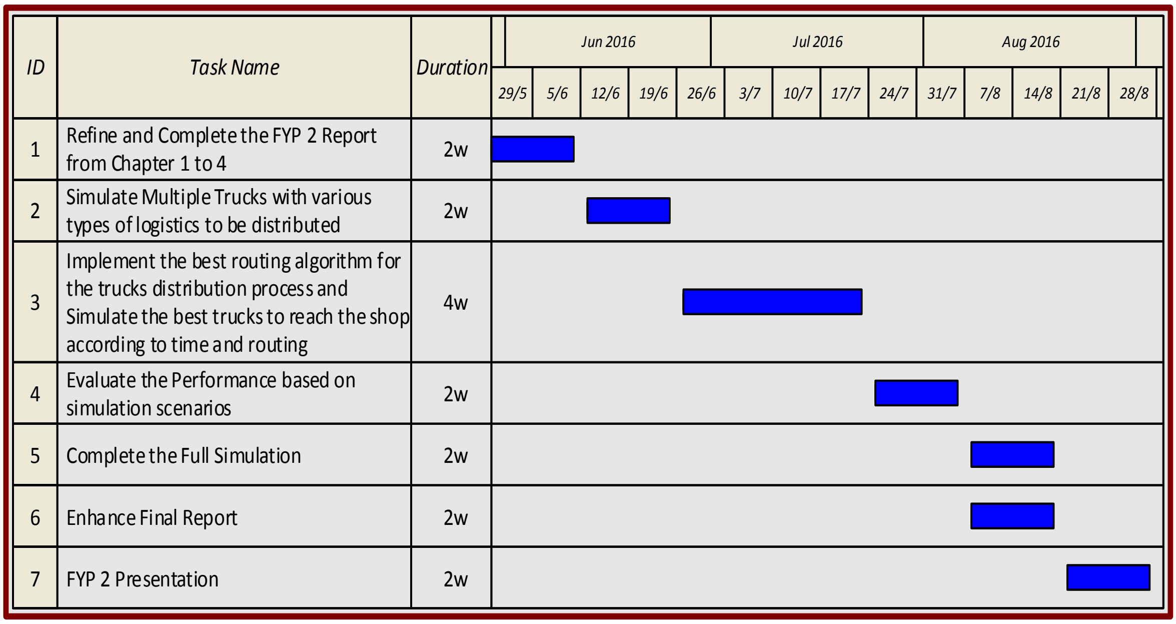


Table 3.3.2: Gantt Chart for FYP 2

3.4 Estimated Cost

In this project, since we are using Simulation of Urban Mobility (SUMO) which is open source as mentioned earlier in the report, and also the computer that used to perform simulation is on my own, hence there are no costs involved in this project.

3.5 Concluding Remark

In this section, the methodology chosen to be applied in this project has been discussed which is Soft System Methodology (SSM). A list of software that will applied to the simulation are described as well and the project milestone is displayed in Gantt Chart as well.

CHAPTER 4: SIMULATION

4.1 Simulation Process

4.1.1 Map Conversion

In SUMO, NETCONVERT, which is the network importer and generator tool in SUMO package, is used to read road networks from different formats and converts them into the SUMO-format so that the map can be used with SUMO simulator.

For this project, the map of the selected city(.osm file) is obtained from Open Street Map which is open source by OpenStreetMap Foundation (OSMF) (OpenStreetMap 2016) with the help of Java Open Street Map. Using NETCONVERT tool, the Open Street Map is then converted to ".net.xml" to be used with SUMO simulator. The following showed the differences after ".osm" file is converted to ".net.xml" using NETCONVERT:

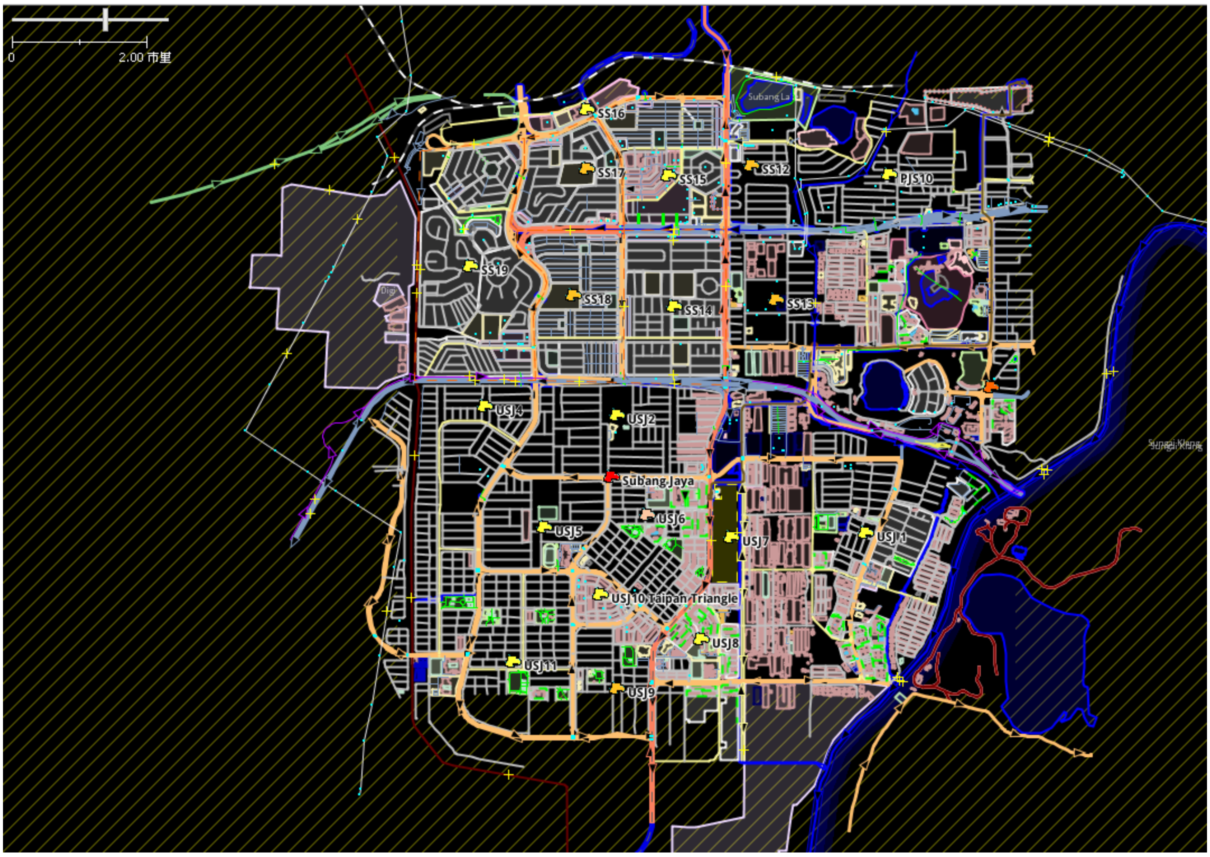


Figure 4.1.1.1: subangjaya.osm

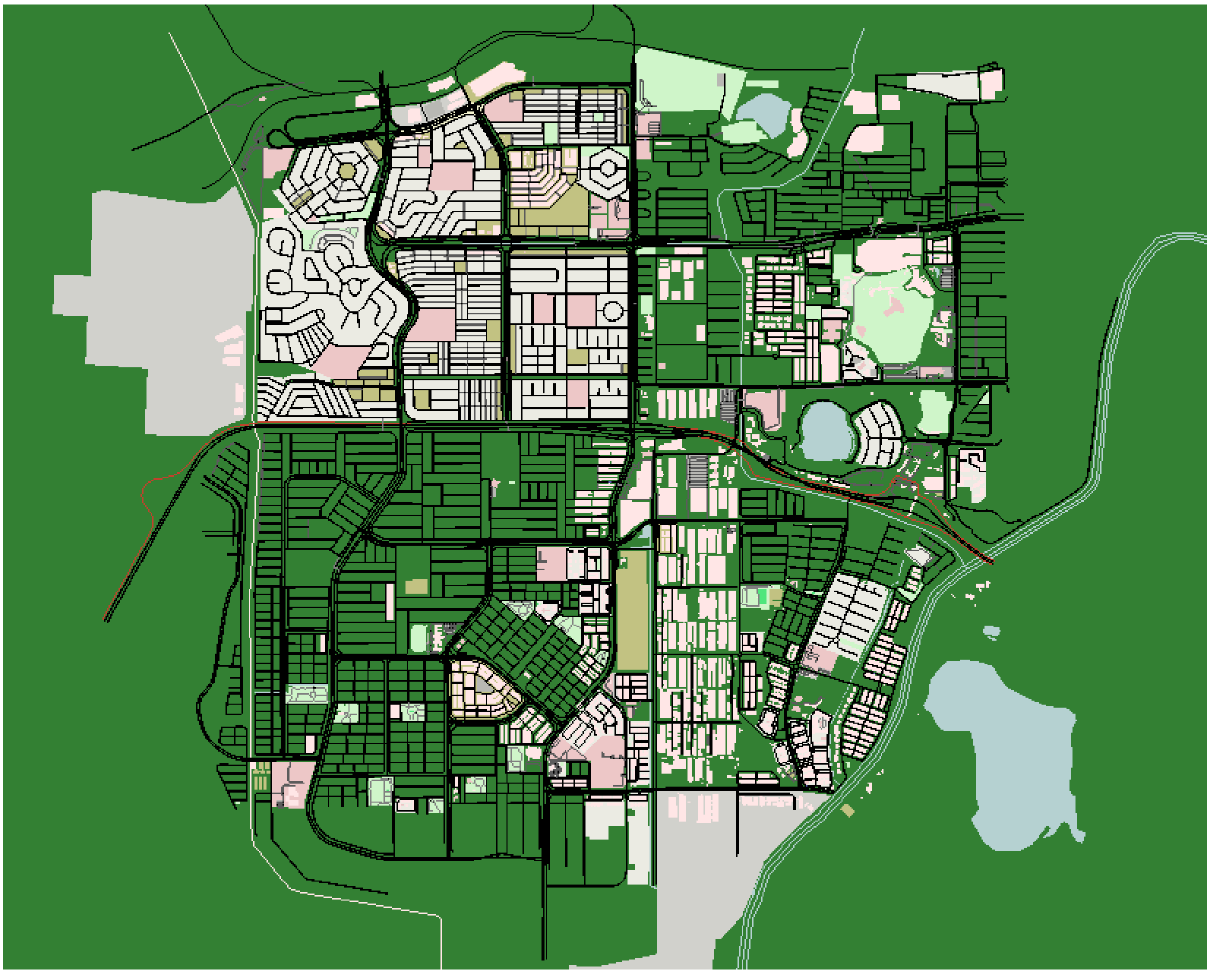


Figure 4.1.1.2: subangjaya.net.xml

4.1.2 Trips Generation

After the network is generated, traffic demand is necessary to have vehicles running on the map. In order to do so, we need trip info and also the route info. To generate route info, we need to have trip info on hand first. Trip contains the info for vehicle movement from one destination to another by defining the starting edge(street), the destination edge and the departure time.

Since this project is simulating a city and it will be complicated to generate the route file from trip file. In order to do so, randomTrips.py which is another tool provided by SUMO is used to generate a set of random network.

The following shows the command to generate the random trips using randomTrips.py where by random vehicles be distributed randomly on their starting edges and inserted with high speed on a reasonable lane:

randomTrips.py -n subangjaya.net.xml --trip-attributes="departLane=\"best\" departSpeed=\"max\" departPos=\"random\"

4.1.3 Routes Generation

In order to generate vehicle movement, we need to combine the .trips file and .net file and parse it through DUAROUTER tool in SUMO. DUAROUTER is a tool used for building vehicle routes from demand definitions, computing routes during a user assignment and repairing connectivity problems in existing route files. Besides that, the vehicle routes are also computed using shortest path computation. There are four routing algorithms available in DUAROUTER which are dijkstra, astar, Contraction Hierarchies, CHWrapper algorithms. The output file is .rou file.

4.1.4 Configuration

When we have net-file and route-file together, we can glue them into configuration file with gui-setting-file to produce .sumocfg file to start simulate the network.

4.1.5 Overview of Simulation Process and Output

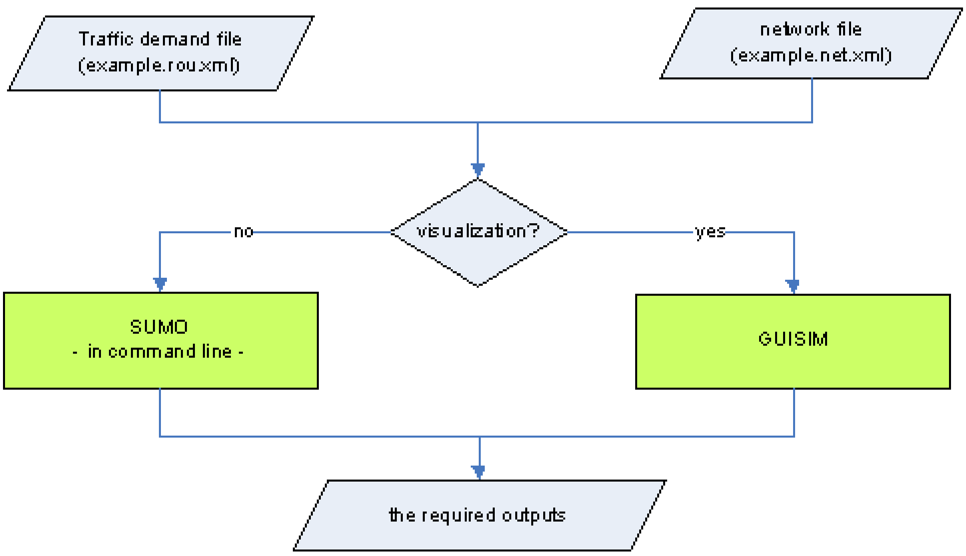


Figure 4.1.5.1: Overview of simulation process

Figure 4.1.5.2: Zoomed in Network with traffic demand

4.1.6 Algorithms for Trucks Backup

In order to assign trucks backup when the principal trucks unable to cover the destination shops at certain time threshold, the logic has to implement in TraaS – Traci as a Service which is an extension for the Sumo simulation program.

The logic is as follow:

- get the lane id for each trucks for every timestep they travel
- if (truck's lane id == shop's location) => checkpoint increased by 1 to the maximum of 5 checkpoints for each trucks
- Assuming at timestep 10500, every trucks should covered at least 50% of the shop destination
- if (totalcheckpoint < 50%) => adding backup trucks to cover last 2 shops
- At the same time, principal trucks need to complete the first 3 shops then only travel back to the starting points, leaving the rest of the shops covered by the

backup trucks. Apply the logic as follow only to the principal trucks which have backup trucks ready:
if (checkpoint == 3) => change truck target to ‘starting point’

4.2 Simulation Setup

In this project, there are a total of 3 trucks that will be transporting the containers from their origins to 3 areas in Subang Jaya where each areas will have 5 stops for the trucks to unload the logistics.

The progress flows of the 3 trucks are as follow:

- Truck1, Truck2 and Truck3 are represented with Red, Blue and Green colour.
- Each of them will pick up a total of 16 containers at the starting points and start distributing to their respecting areas where Truck1 is assigned to Area A, Truck2 to Area B and Truck3 to Area C.
- At each area, for example Area A, there will be 5 shops namely Shop A1, Shop A2, Shop A3, Shop A4 and Shop A5. Shop A1 will required 4 containers and the other 12 containers are equally distributed to each shops.
- After the trucks finishing distributing the containers, they will then return to their original destinations.

Figure 4.2.1: A skyview map of the destinations whereby the trucks will be travelling from the starting points.

Truck1: Starting Point -> Shop A1 -> Shop A2 -> Shop A3 -> Shop A4 -> Shop A5 -> Starting Point

Truck2: Starting Point -> Shop B1 -> Shop B2 -> Shop B3 -> Shop B4 -> Shop B5 -> Starting Point

Truck3: Starting Point -> Shop C1 -> Shop C2 -> Shop C3 -> Shop C4 -> Shop C5 -> Starting Point

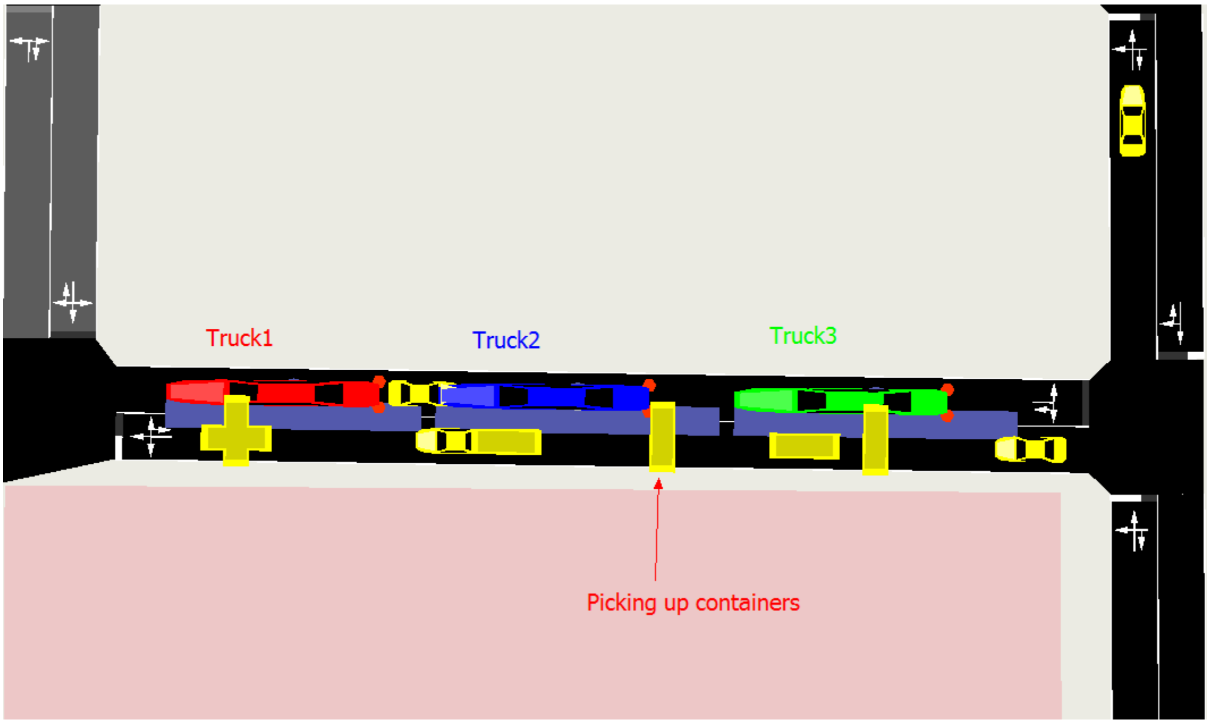


Figure 4.2.2: Showing 3 trucks are picking up the containers at the starting points before transporting them to destined destinations.

4.3 Concluding Remark

This section has described the basic idea on how's the simulation process of SUMO. At the same time, an overview design flows of the transportations of trucks are also illustrated integrating with the map of Subang Jaya.

The implementation of the simulations and the scenarios during the transportations of the trucks will further discussed in Chapter 5.

CHAPTER 5: SIMULATION IMPLEMENTATION

5.1 Setting and Configuration

5.1.1 Routing Configurations

By default, SUMO is using Dijkstra algorithm to run the simulation. Besides Dijkstra routing algorithm, there are Astar and Contraction Hierachy routing methods available as well so that the best algorithm will be chosen for the current map approach.

The routing configuration can be defined in the Sumo configuration file as shown:

```
</routing> -->
<routing>
    <device.rerouting.explicit value ="truck1,truck2,truck3,Backup1,Backup2,Backup3"/>
    <device.rerouting.period value = "300"/>
    <routing-algorithm value = 'CH'/>
</routing>
```

Figure 5.1.1.1: Routing Configurations

5.1.2 Truck Re-routing Configurations

Truck Re-routing capability is needed when the vehicles need to adapt to their routes while running so as to avoid time loss during traffic jam. The vehicle ID's need to be defined first so that they have the capability to perform rerouting and then the rerouting period is set to 300s so that they are able to calculate the new routes every 300s.

The configurations are done in Sumo configuration file as well.

```
<routing>
    <device.rerouting.explicit value ="truck1,truck2,truck3,Backup1,Backup2,Backup3"/>
    <device.rerouting.period value = "300"/>
</routing>
```

Figure 5.1.2.1: Truck Re-routing Configurations

5.1.3 Logistics and Container Stops Configurations

The Container Stops are configured so as to determine the locations where the trucks will transport their logistics. The configurations need to be setup in additional files as follow:

```
                                   Starting Points
<containerStop id="Station1" lane="-25717935#2_0" startPos="5" endPos="25"  lines="Station1" />
<containerStop id="Station2" lane="-25717935#2_0" startPos="26" endPos="46"  lines="Station2" />
<containerStop id="Station3" lane="-25717935#2_0" startPos="47" endPos="65"  lines="Station3" />
                                   Shops in Area A
<containerStop id="StopA2" lane="53655489#0_0" startPos="30" endPos="50"  lines="Shop A1" />
<containerStop id="StopA3" lane="-102562074_0" startPos="30" endPos="50"  lines="Shop A2" />
<containerStop id="StopA4" lane="410537373#3_0" startPos="30" endPos="50"  lines="Shop A3" />
<containerStop id="StopA5" lane="-221195042#2_0" startPos="30" endPos="50"  lines="Shop A4" />
<containerStop id="StopA6" lane="-211071550_0" startPos="10" endPos="30"  lines="Shop A5" />
                                   Shops in Area B
<containerStop id="StopB2" lane="-53633208#0_0" startPos="30" endPos="50"  lines="Shop B1" />
<containerStop id="StopB3" lane="220002833_0" startPos="30" endPos="50"  lines="Shop B2" />
<containerStop id="StopB4" lane="-32919463#1_0" startPos="10" endPos="30"  lines="Shop B3" />
<containerStop id="StopB5" lane="222423130_0" startPos="100" endPos="130"  lines="Shop B4" />
<containerStop id="StopB6" lane="-53658166#2_0" startPos="10" endPos="30"  lines="Shop B5" />
                                   Shops in Area C
<containerStop id="StopC2" lane="-48951015#1_0" startPos="0" endPos="20"  lines="Shop C1" />
<containerStop id="StopC3" lane="-222782965#5_0" startPos="30" endPos="50"  lines="Shop C2" />
<containerStop id="StopC4" lane="-222972257#0_0" startPos="30" endPos="50"  lines="Shop C3" />
<containerStop id="StopC5" lane="222915568#7_0" startPos="30" endPos="50"  lines="Shop C4" />
<containerStop id="StopC6" lane="223637652#1_0" startPos="30" endPos="50"  lines="Shop C5" />
```

Figure 5.1.3.1: Containers configurations

Next, the shops that will be travelled by the trucks need to be defined as follow:

```
<vehicle id="truck1" type="truck" depart="0" color="red" >
    <route edges="-25717935#2 25717935#2 -25717908#2 -25717908#1 25718014#24 25718014#25 25718014#26 102754096 402934743#4 402934743#4-AddedOffRamp
        <stop containerStop="Station3" duration="100" parking="true" />
        <stop containerStop="StopA2" duration="100" parking="true" />
        <stop containerStop="StopA3" duration="100" parking="true" />
        <stop containerStop="StopA4" duration="100" parking="true" />
        <stop containerStop="StopA5" duration="100" parking="true" />
        <stop containerStop="StopA6" duration="100" parking="true" />
    </route>
</vehicle>
```

Figure 5.1.3.2: Containers stops configurations

For the backup trucks travelling locations, only the last 2 stations are needed since when the trucks in charged couldn’t cover more than 50% of the stations at certain time threshold, the backup trucks will cover the rest of the stations. In the scenario here, there are 5 stations initially needed to be covered by the principal trucks, if the time threshold has been reached, the backup trucks will be triggered and cover the last 2 stations for each areas.

```
<!-- for backup1 purple  -->
<vType id="truckbackup"  accel="2.6" decel="4.5" sigma="0.5" length="15" maxSpeed="100"
        color="153,51,235" loadingDuration="10.0" containerCapacity="50"/>

<route id="B1" edges="-25717935#2 -25717935#0 -25718014#19 -25718014#18 -5202764#0 52218328#3 52218328#3-AddedOffRampEdge 25016243#0 25016243#1 19
  <stop containerStop="Station3" duration="100" parking="true" />
  <stop containerStop="StopA5" duration="100" parking="true" />
  <stop containerStop="StopA6" duration="100" parking="true" />
</route>
<!-- for backup1 purple  -->

<!-- for backup2 orange  -->
<vType id="truckbackup2"  accel="2.6" decel="4.5" sigma="0.5" length="15" maxSpeed="100"
        color="255,128,0" loadingDuration="10.0" containerCapacity="50"/>

<route id="B2" edges="-25717935#2 -25717935#0 -25718014#19 -25718014#18 -5202764#0 52218328#3 52218328#3-AddedOffRampEdge 52218328#4 103299427#1-A
  <stop containerStop="Station2" duration="100" parking="true" />
  <stop containerStop="StopB5" duration="100" parking="true" />
  <stop containerStop="StopB6" duration="100" parking="true" />
</route>
<!-- for backup2 orange  -->
```

Figure 5.1.3.3: Backup trucks configurations

The number of containers that will be carried by the trucks is defined in the route file. For instance, if there are 4 containers required to be transported to Shop A1 which is located on the edge with id “53655489#0” by the truck1, there are a total of 4 container id need to be specified from the same starting point to the same edge id as shown:

```
<routes xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance" xsi:noNamespaceSchemaLocat

    <container id="containerA_1" depart="0" departPos="60" >
        <transport from="-25717935#2" to="53655489#0" lines="truck1"/>
    </container>
    <container id="containerA_2" depart="0" departPos="60" >
        <transport from="-25717935#2" to="53655489#0" lines="truck1"/>
    </container>
    <container id="containerA_3" depart="0" departPos="60" >
        <transport from="-25717935#2" to="53655489#0" lines="truck1"/>
    </container>
    <container id="containerA_4" depart="0" departPos="60" >
        <transport from="-25717935#2" to="53655489#0" lines="truck1"/>
    </container>

    <container id="containerA_5" depart="0" departPos="60" >
        <transport from="-25717935#2" to="410537373#3" lines="truck1"/>
    </container>
    <container id="containerA_6" depart="0" departPos="60" >
        <transport from="-25717935#2" to="410537373#3" lines="truck1"/>
    </container>
    <container id="containerA_7" depart="0" departPos="60" >
        <transport from="-25717935#2" to="410537373#3" lines="truck1"/>
    </container>
```

Figure 5.1.3.4: Containers Transportation Configurations

5.1.4 Traas Setup in Java Eclipse

Traas is needed for scenario based simulation whereby the backup trucks are called upon when the principal trucks unable to cover at least 50% of the shops at certain time threshold. In this project, a timestep of ‘10500’ is assumed and set as the threshold whereby all the principal trucks should cover at least 50% of their destined shops.

The backup trucks are called upon based on the logic as follow:

When the specific truck reach the destined shop, the checkpoint will then be recorded.

```
if(count<checkpointno)
{
    String laneid = (String)conn.do_job_get(Vehicle.getLaneID("truck1"));

     if (laneid.equals("53655489#0_0") && checked== false)              Shop A1
     {
         checkpoint = checkpoint+1;
         count = count + 1;
         System.out.println("checkpoint 53655489#0_0: " + checkpoint);
         System.out.println("Time: " + i);
         checked = true;

     }

     if (laneid.equals("-102562074_0")&& checked== true)
     {
         checkpoint = checkpoint+1;                                     Shop A2
         count = count + 1;
         System.out.println("checkpoint -102562074_0: " + checkpoint);
         System.out.println("Time: " + i);
         checked = false;
     }

     if (laneid.equals("410537373#3_0")&& checked== false)
     {                                                                  Shop A3
         checkpoint = checkpoint+1;
         count = count + 1;
         System.out.println("checkpoint 410537373#3_0: " + checkpoint);     ...
         System.out.println("Time: " + i);
         checked = true;
     }
```

Figure 5.1.4.1: Java algorithms implementation

At a timestep which assumed to be “10500s”, the percentage of the shops covered are calculated based on the checkpoints recorded. When the checkpoints covered do not meet 50%, backup trucks are assigned in order to cover the last 2 shops for each areas so as to reduce the burden of the principal trucks. There are a total of 3 backup trucks ready for each trucks.

The principal trucks still have to cover the first 3 shops while the rest of the destinations are covered by the backup trucks.

Timestep Threshold

Calculate the percentage of shops covered

Assign trucks to cover the shops if percentage of principal trucks do not meet 50%

```
if(i==10500)
{
    double checkpointcovered1=(checkpoint/checkpointno)*100;
    double checkpointcovered2 = (checkpoint2/checkpointno)*100;
    double checkpointcovered3 = (checkpoint3/checkpointno)*100;

    System.out.println("Checkpoint Covered for truck1: " + checkpointcovered1 + "%");
    System.out.println("Checkpoint Covered for truck2: " + checkpointcovered2 + "%");
    System.out.println("Checkpoint Covered for truck3: " + checkpointcovered3 + "%");

    if(checkpointcovered1<50)
    {
        System.out.println("Backup1 truck ready");
        conn.do_job_set(Vehicle.add("Backup1","truckbackup","B1",i,50,0,(byte) -2));
        backup1 = true;

    }

    if(checkpointcovered2<50)
    {
        System.out.println("Backup2 truck ready");
        conn.do_job_set(Vehicle.add("Backup2","truckbackup2","B2",i,30,0,(byte) -2));
        backup2 = true;

    }

    if(checkpointcovered3<50)
    {
        System.out.println("Backup3 truck ready");
        conn.do_job_set(Vehicle.add("Backup3","truckbackup3","B3",i,5,0,(byte) -2));
        backup3 = true;

    }
}
```

Figure 5.1.4.2: Java algorithms implementation

After the principal trucks cover the first 3 shops, they are required to go back to the starting points if the backup trucks are assigned to cover the last 2 shops.

```java
if(backup1==true && checkpoint==3)
{
    System.out.println("Calling Truck1 Stops operation");
    conn.do_job_set(Vehicle.changeTarget("truck"+1, "-25717935#2"));
    backup1 = false;
    count = 5; //stop checking checkpoint
}

if(backup2==true && checkpoint2==3)
{
    System.out.println("Calling Truck2 Stops operation");
    conn.do_job_set(Vehicle.changeTarget("truck"+2, "-25717935#2"));
    backup2 = false;
    count2 = 5; //stop checking checkpoint
}

if(backup3==true && checkpoint3==3)
{
    System.out.println("Calling Truck3 Stops operation");
    conn.do_job_set(Vehicle.changeTarget("truck"+3, "-25717935#2"));
    backup3 = false;
    count3 = 5; //stop checking checkpoint
}
```

Call Back to Starting Points

Figure 5.1.4.3: Java algorithms implementation

5.2 Simulation Operations

5.2.1 Trucks Transportation Simulation

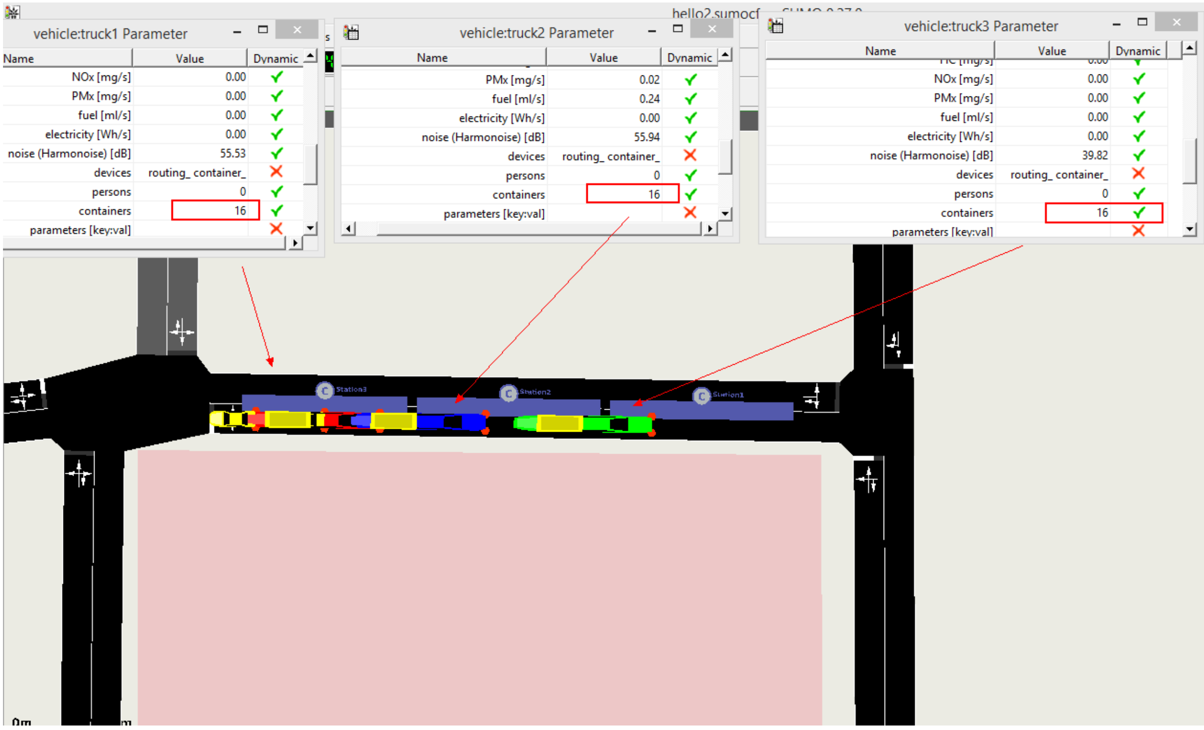


Figure 5.2.1.1: In the beginning of the simulation, 3 trucks are picking up a total of 16 containers each from the starting point.

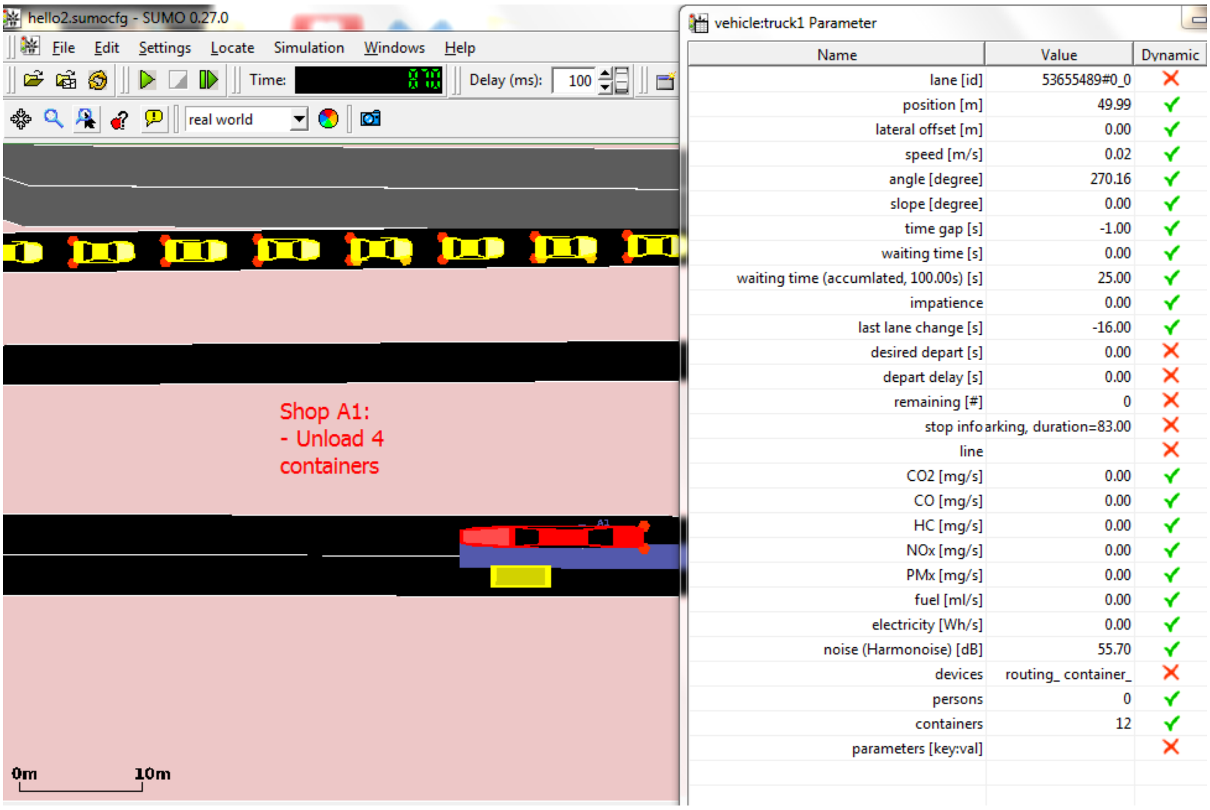


Figure 5.2.1.2: Taking Truck1 as example, after it reached Shop A1 which is the first stop, it will unload a total of 4 containers so that the remaining containers left on the trucks are 12.

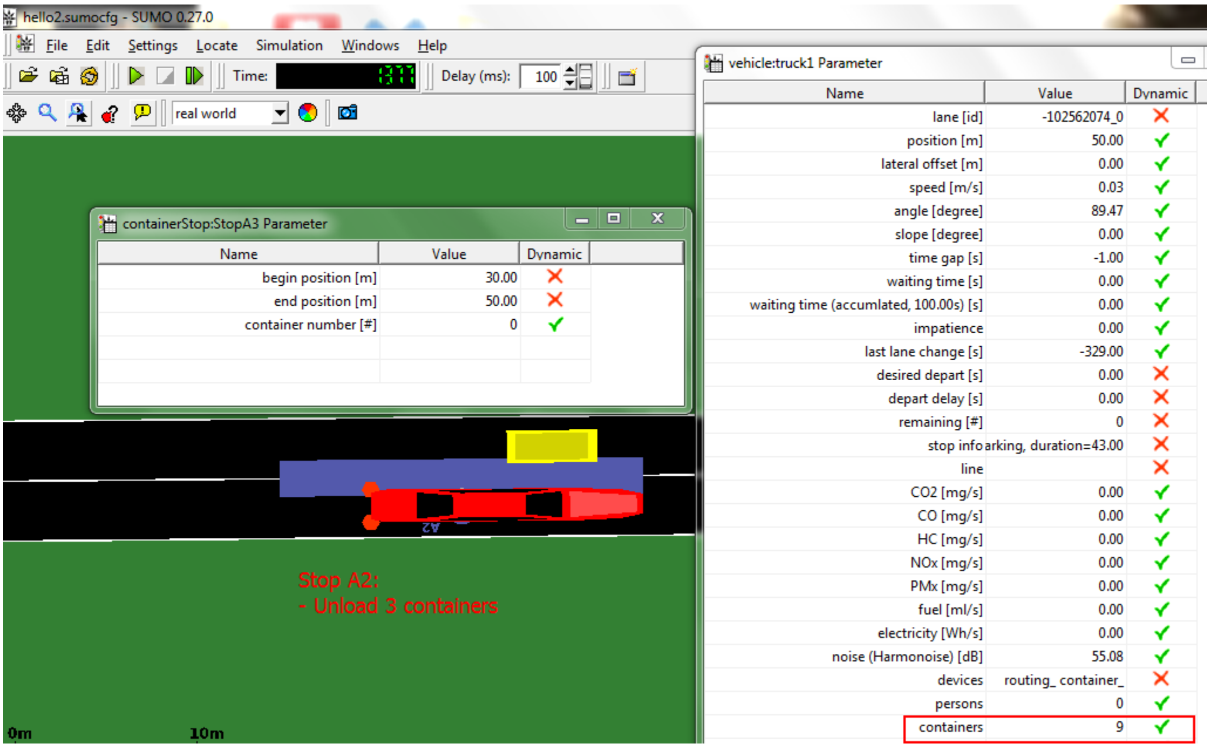


Figure 5.2.1.3: Reaching Shop A2 and after unloading 3 containers, there are 9 containers remaining on the truck1

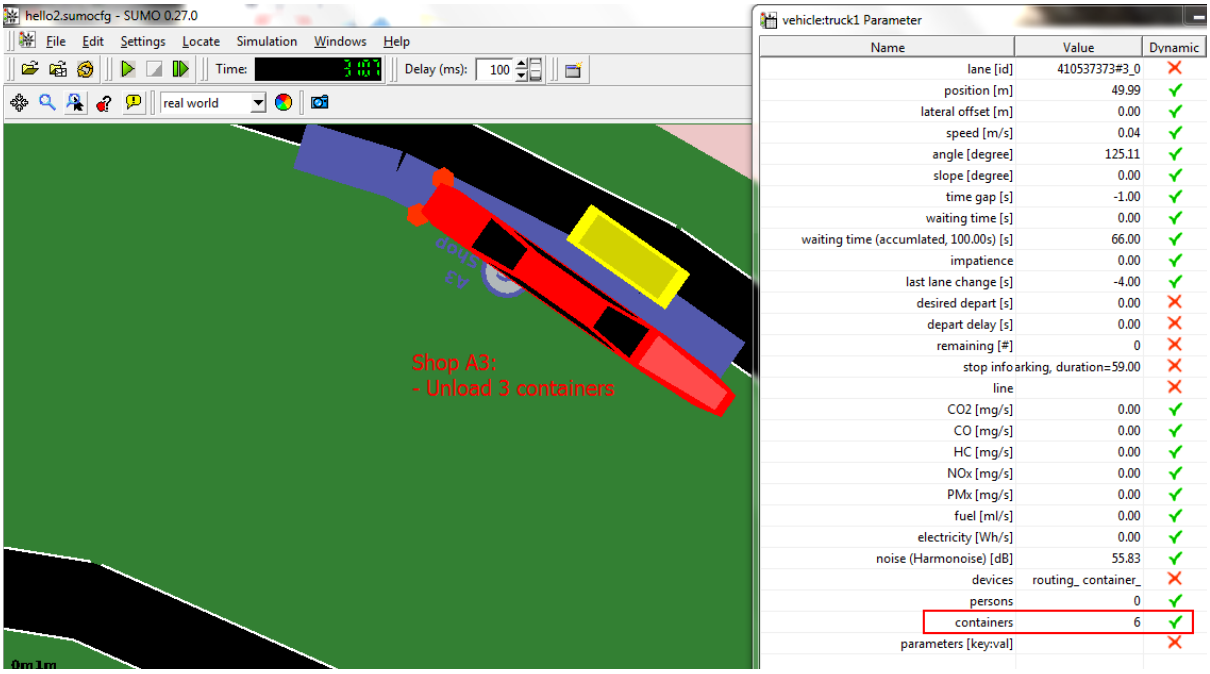


Figure 5.2.1.4: At Shop A3, 3 containers are unloaded and there are remaining 6 containers

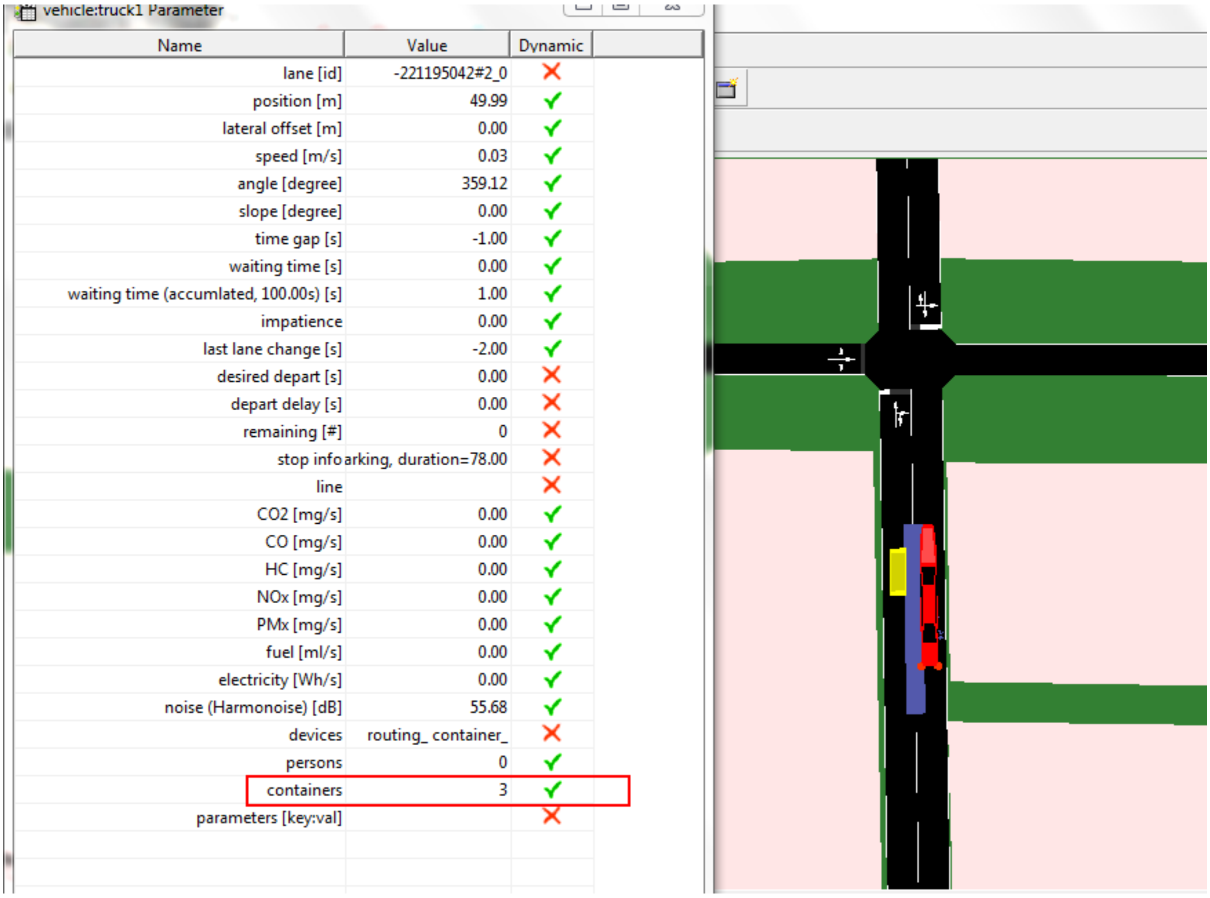


Figure 5.2.1.5: Reaching Shop A4 and unloading 3 containers as well, the remaining containers left are 3

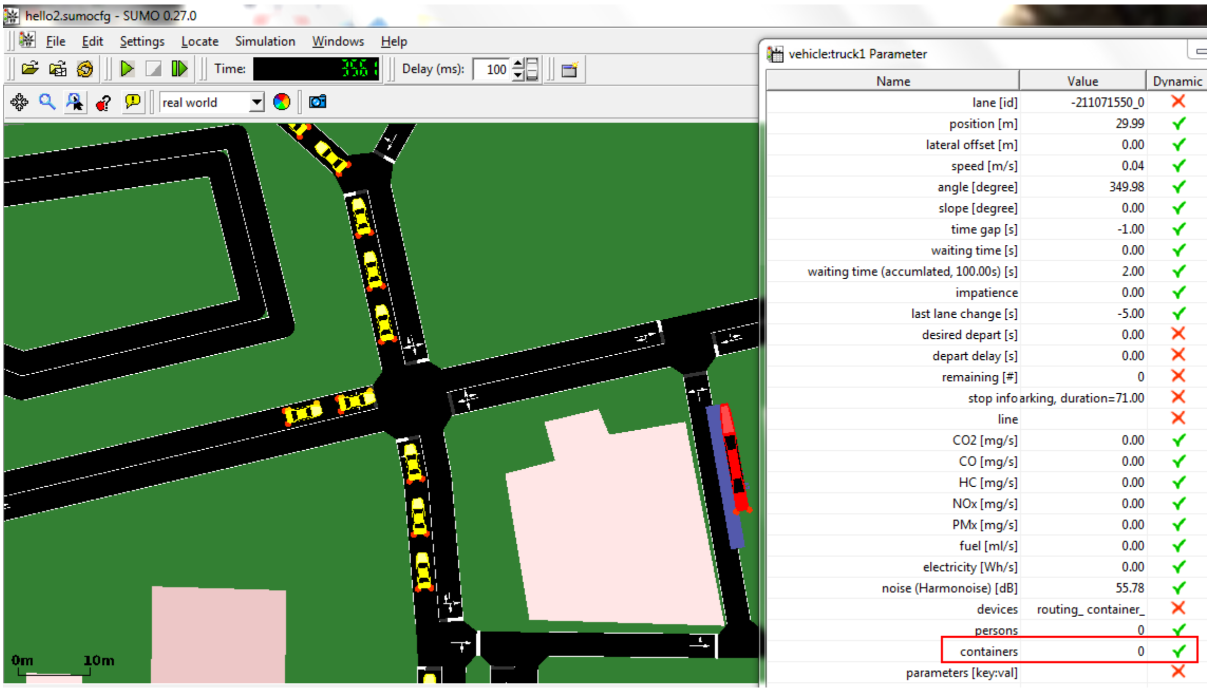


Figure 5.2.1.6: At Shop A5 which is the last distribution destination, no more containers are left and truck1 finished it's task according to the plan.

5.2.2 Trucks Automatic Routing Capabilities Simulation

This section will be divided into 2 parts to illustrate the differences with or without automatic rerouting capabilities on the trucks. The trucks with rerouting capability are configured to reroute themselves for every 300s

5.2.2.1 Trucks without Automatic Routing Capabilities

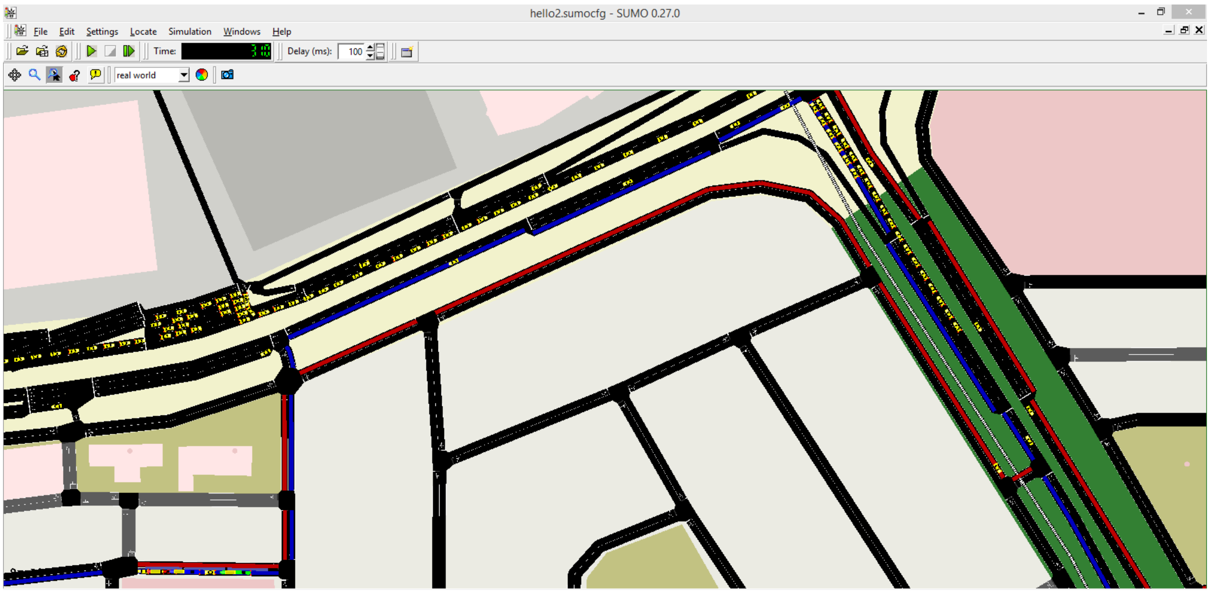


Figure 5.2.2.1.1: Simulation started without automatic routing capabilities on trucks

When the simulation started, the lanes that will be covered by the trucks are represented with their respective colours, whereby red strip is for truck1, blue strip for truck2 and green strip(not shown in the figure) is for truck3

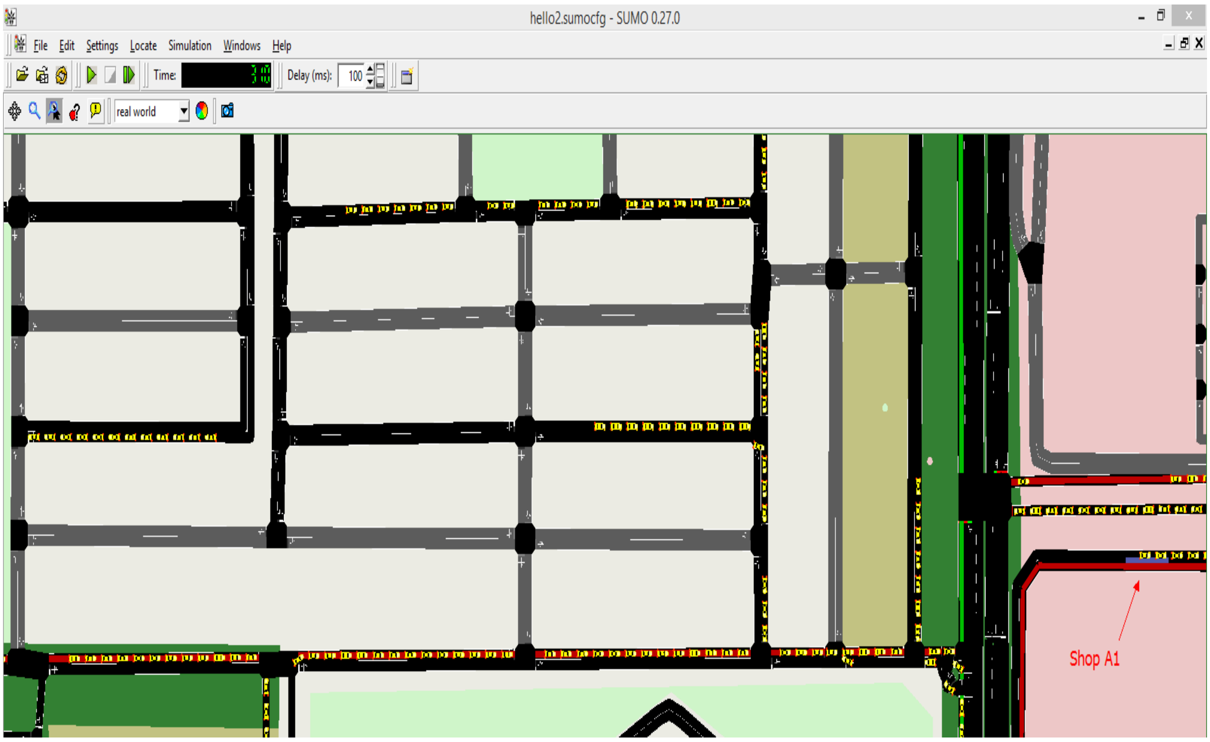


Figure 5.2.2.1.2: Route to be travelled by Truck1

From the figure above, observe that the lane covered by truck1 is quite jam at the timesteps of 310.

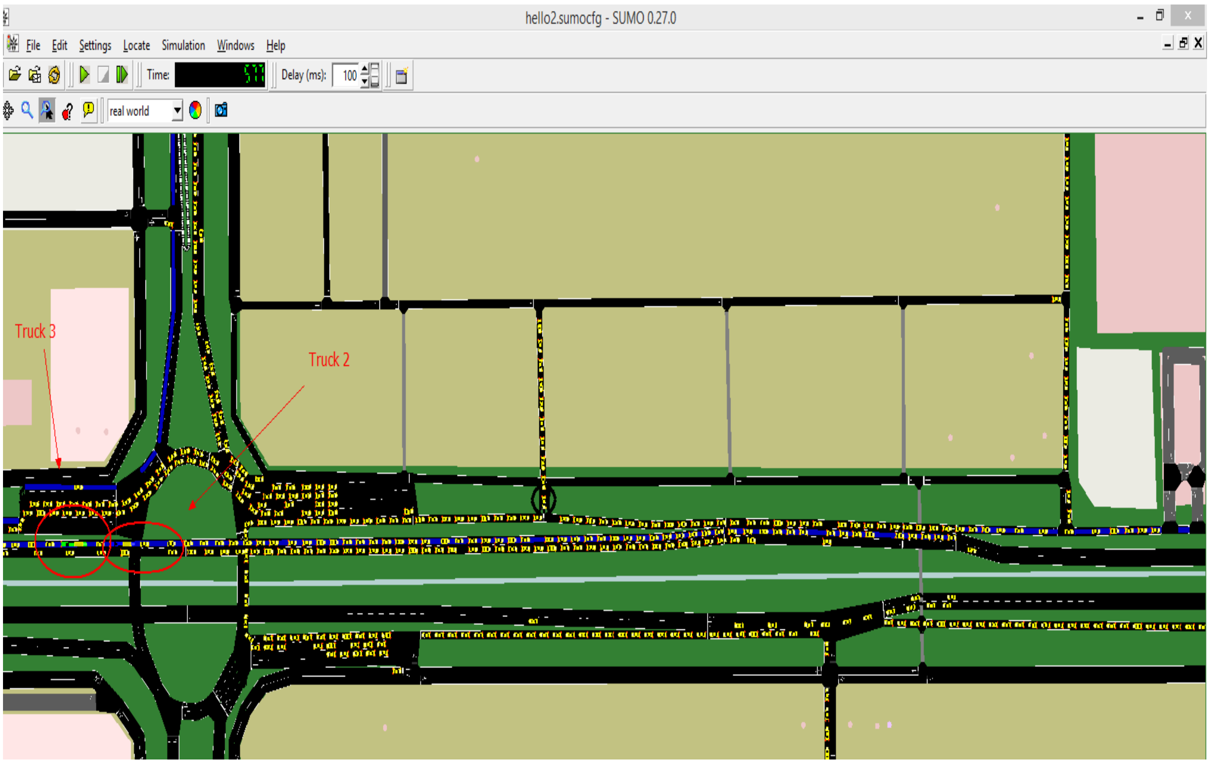


Figure 5.2.2.1.3: Situation of Truck2 and Truck3 without automatic routing capabilities

From the figure above, it is observed that Truck2 and Truck3 are stuck in the traffic jam at the timesteps of 577.

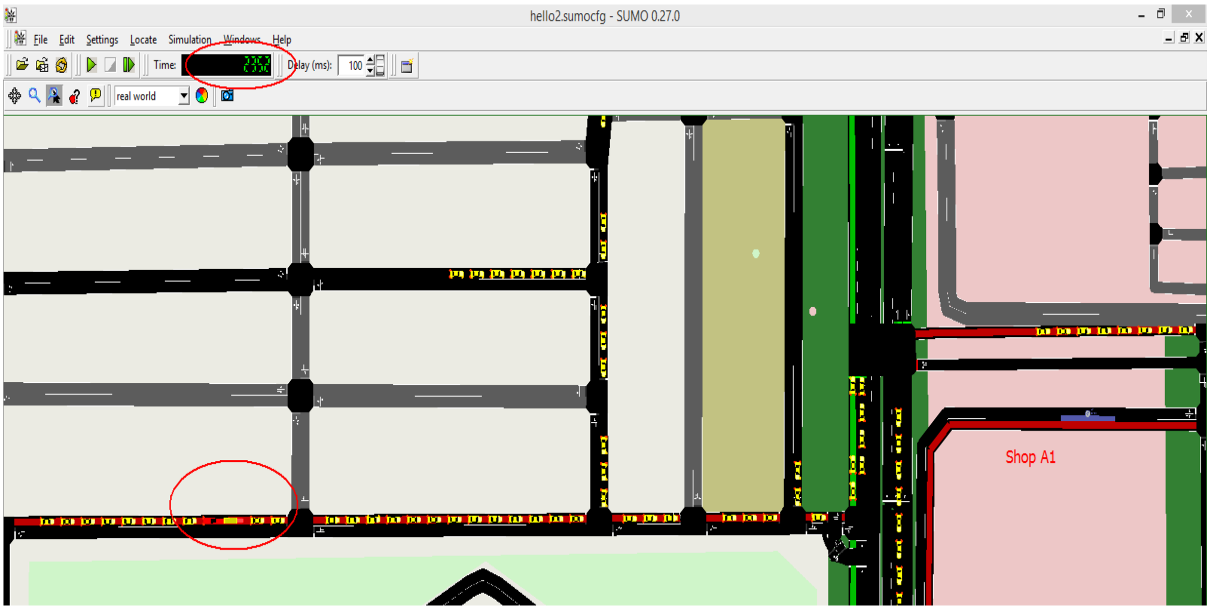


Figure 5.2.2.1.4: Situation of Truck1 at timestep 2352

When reaching timestep of 2352, Truck1 is still stuck in the traffic and yet transport the containers to Shop A1

5.2.2.2 Trucks with Automatic Routing Capabilities

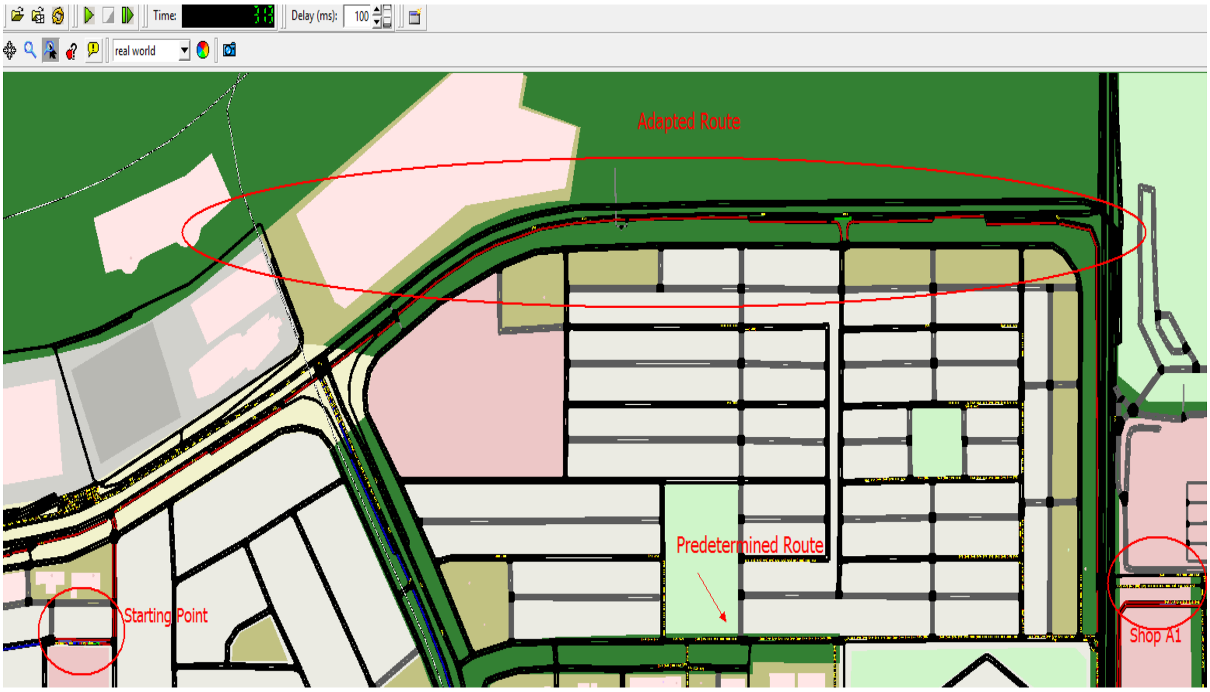


Figure 5.2.2.2.1: Adapted route of Truck1 with automatic routing capability

At timestep of 310s where the automatic routing capability of Truck1 is triggered, it can observed that the predetermined route of Truck1 has been changed to a lane which is less traffic when comparing to Figure 5.2.2.1.2

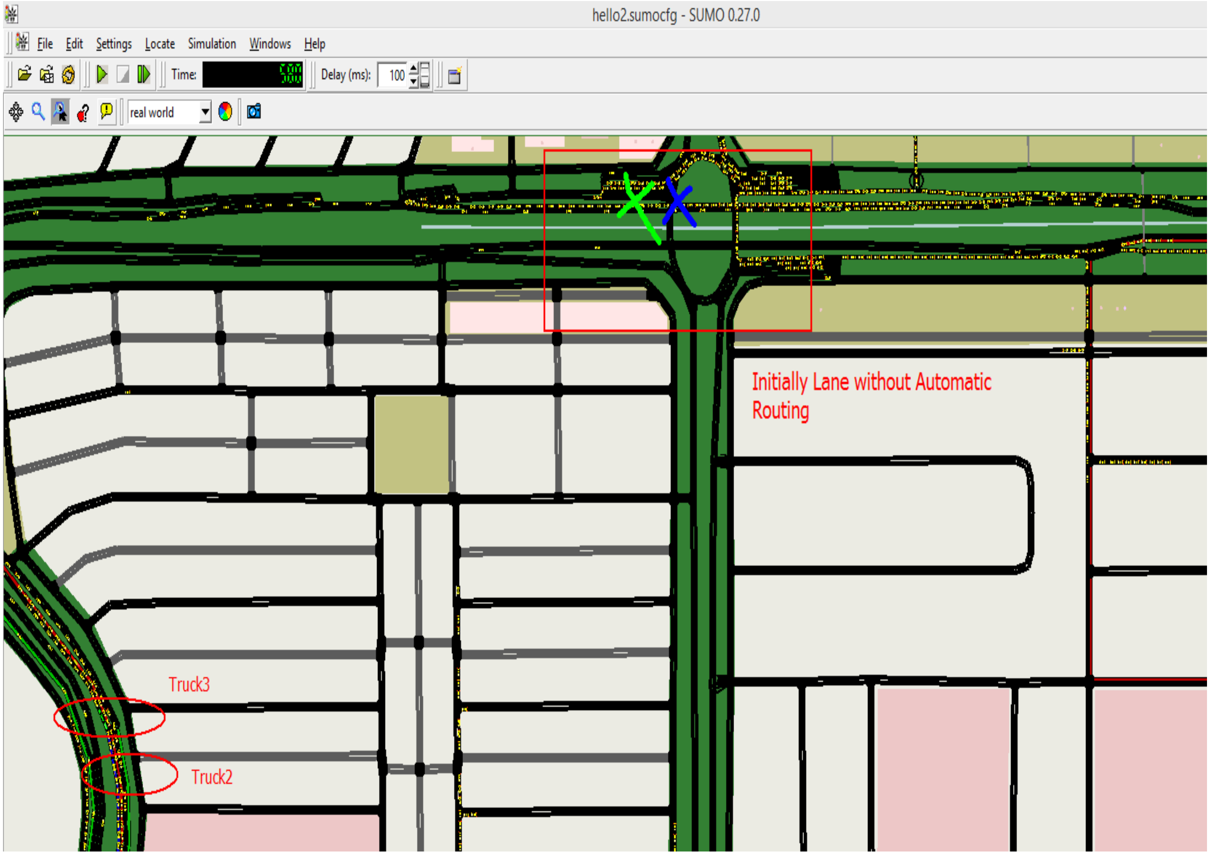


Figure 5.2.2.2.2: Situation of Truck2 and Truck3 with automatic routing capabilities

Comparing to Figure 5.2.2.1.3, it is observed that Truck3 and Truck2 are having different routes after calculating the best path.

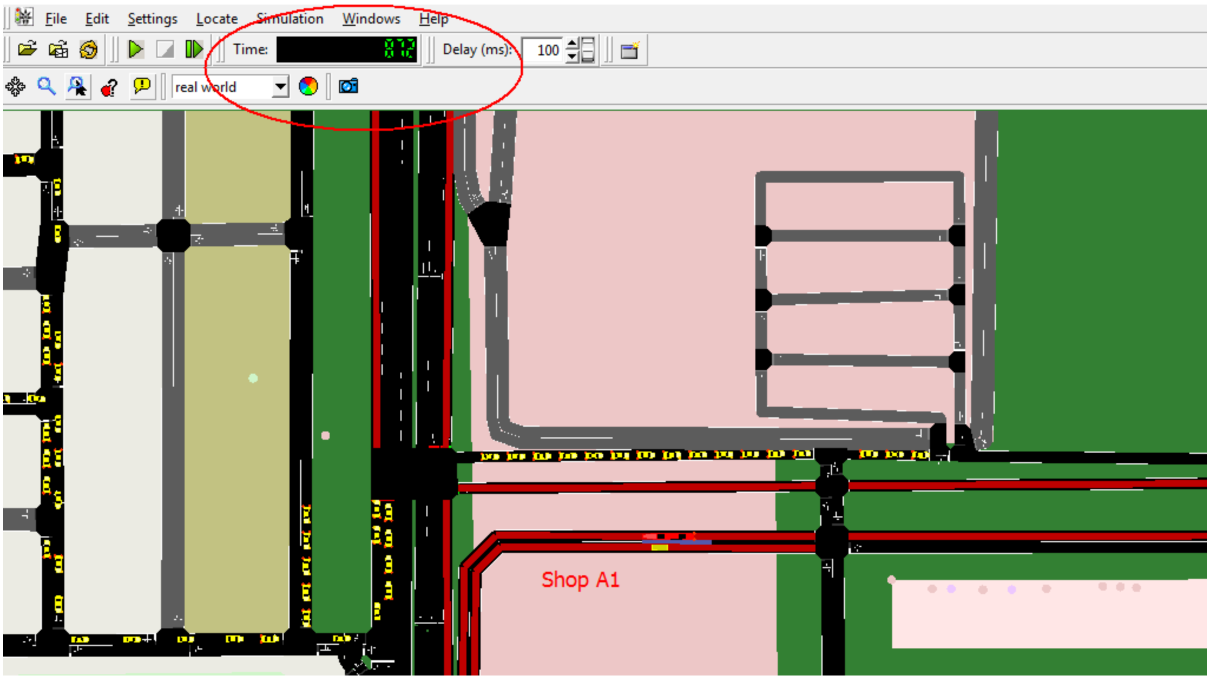


Figure 5.2.2.2.3: Situation of Truck1 with automatic routing capability

Comparing to Figure 5.2.2.1.4, at timestep of 872s, Truck1 with automatic routing capability already showed its advantage by reaching Shop A1 at a faster time pace comparing to the scenario where the truck1 is without automatic routing capability and still on the way to Shop A1 at timestep of 2352s.

5.2.3 Scenario Based – Assigning Backup Trucks

In this project, backup trucks are called upon when the principal trucks unable to cover at least 50% of the shops within the timestep of 10500 as stated in section 5.1.4.

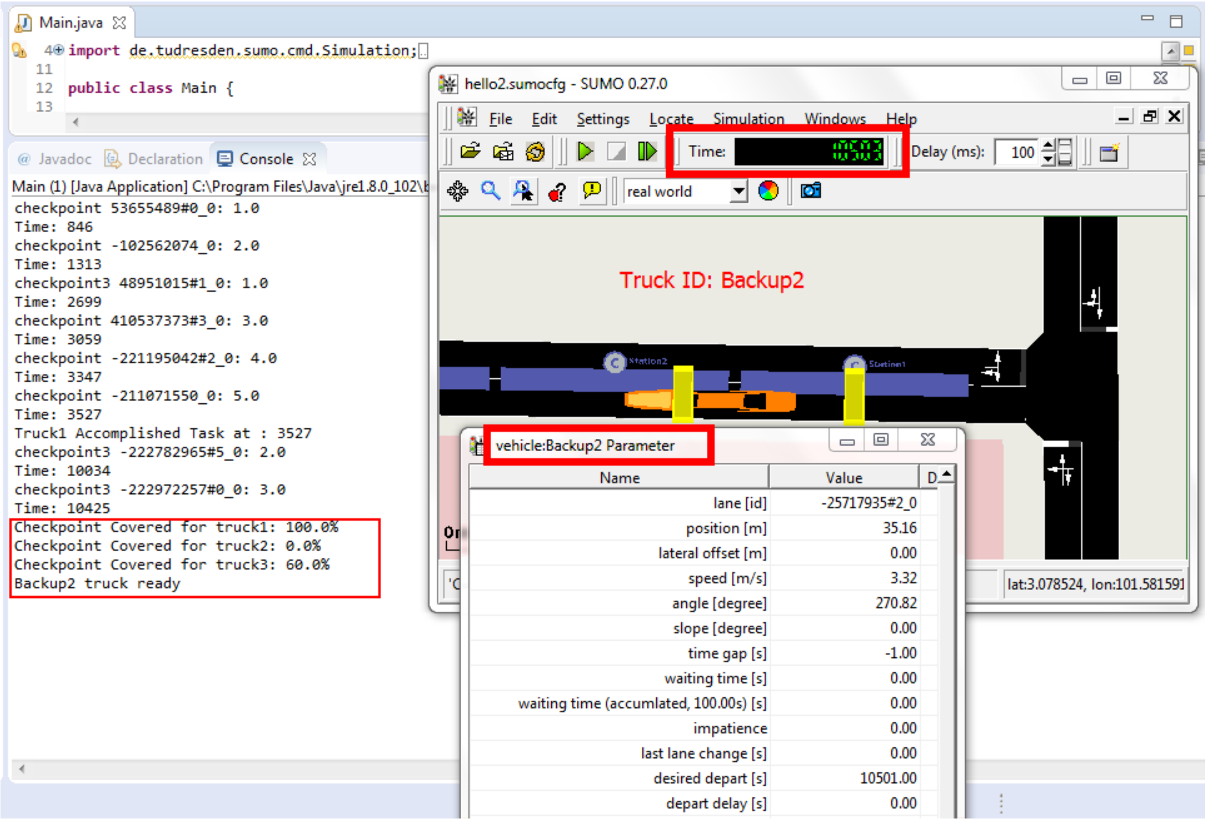


Figure 5.2.3.1: Truck ID Backup2 is called from the starting point

When the simulation reaching a timestep of 10500s which is defined in the program, the percentage of checkpoints covered by the trucks are displayed. Since Truck2 doesn’t reach the requirement of 50% coverage, the backup truck with ID: Backup2 is called to cover the last 2 destination of Truck2 whereas Truck2 will still need to cover the first 3 shops which are Shop B1, B2 and B3.

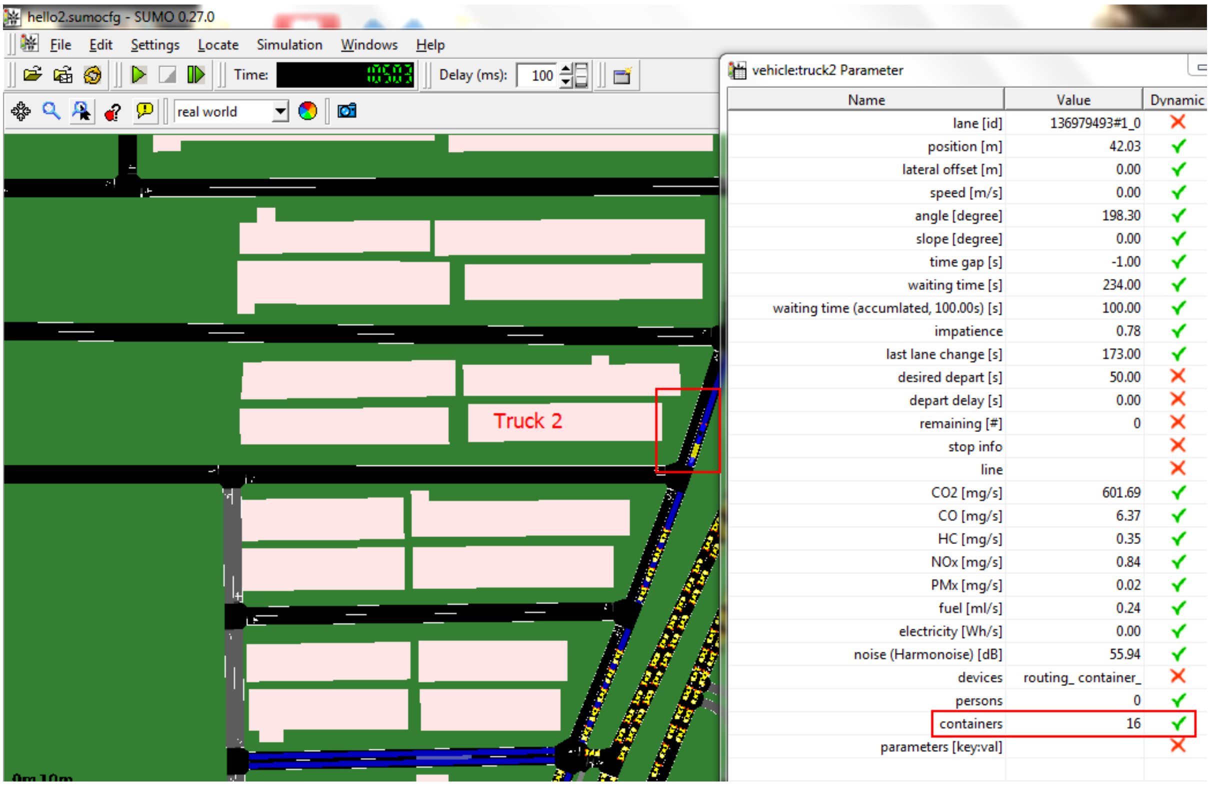


Figure 5.2.3.2: Current situation Truck2 which fall in traffic jam

As seen from the figure above, Truck2 is still in the traffic and none of the containers been distributed. This has proved that the checkpoints covered is only 0% as shown in Figure 5.2.3.1 which triggered Backup2 truck.

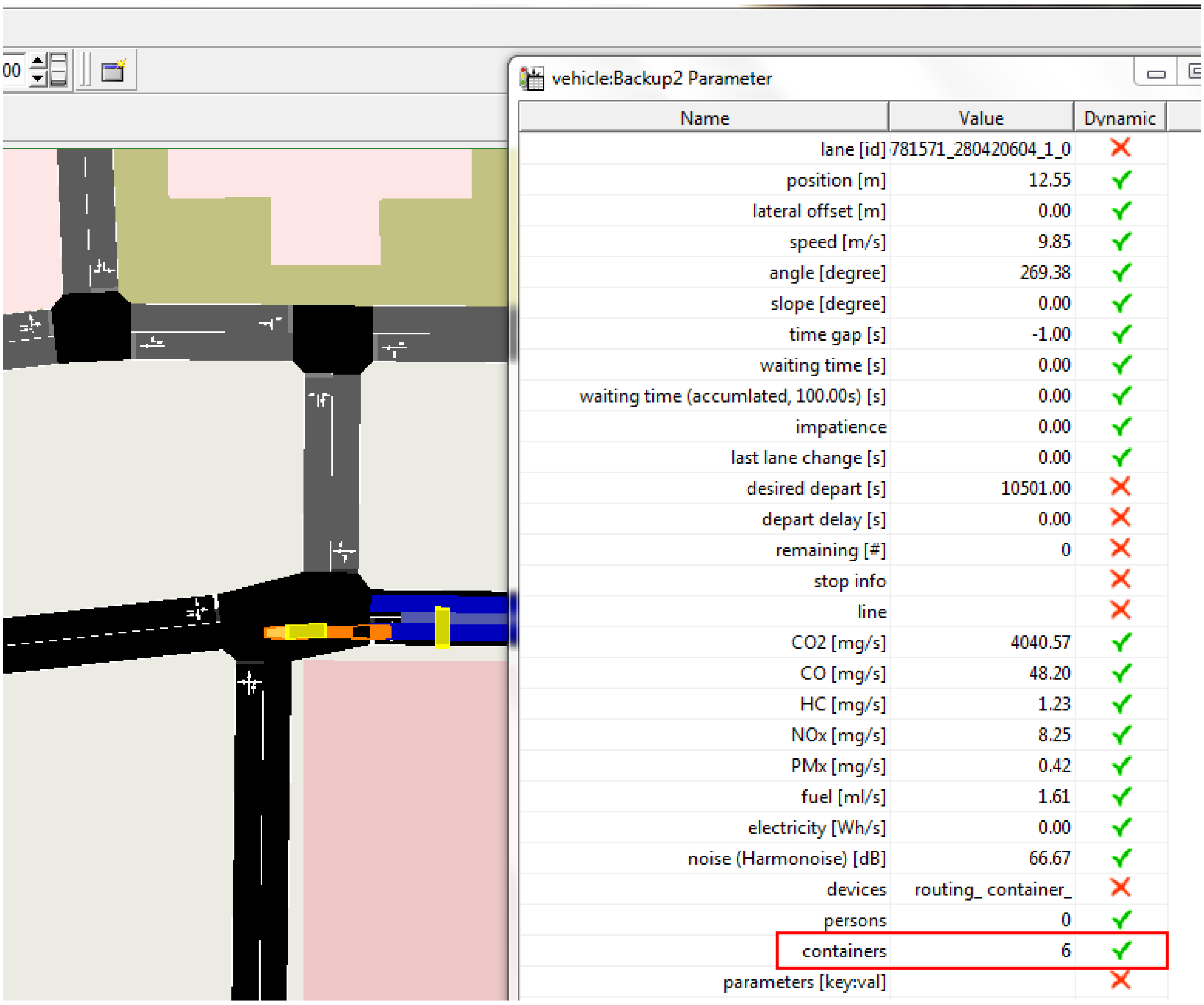


Figure 5.2.3.3: Backup2 Truck pick up 6 containers which are required by Shop B4 and Shop B5

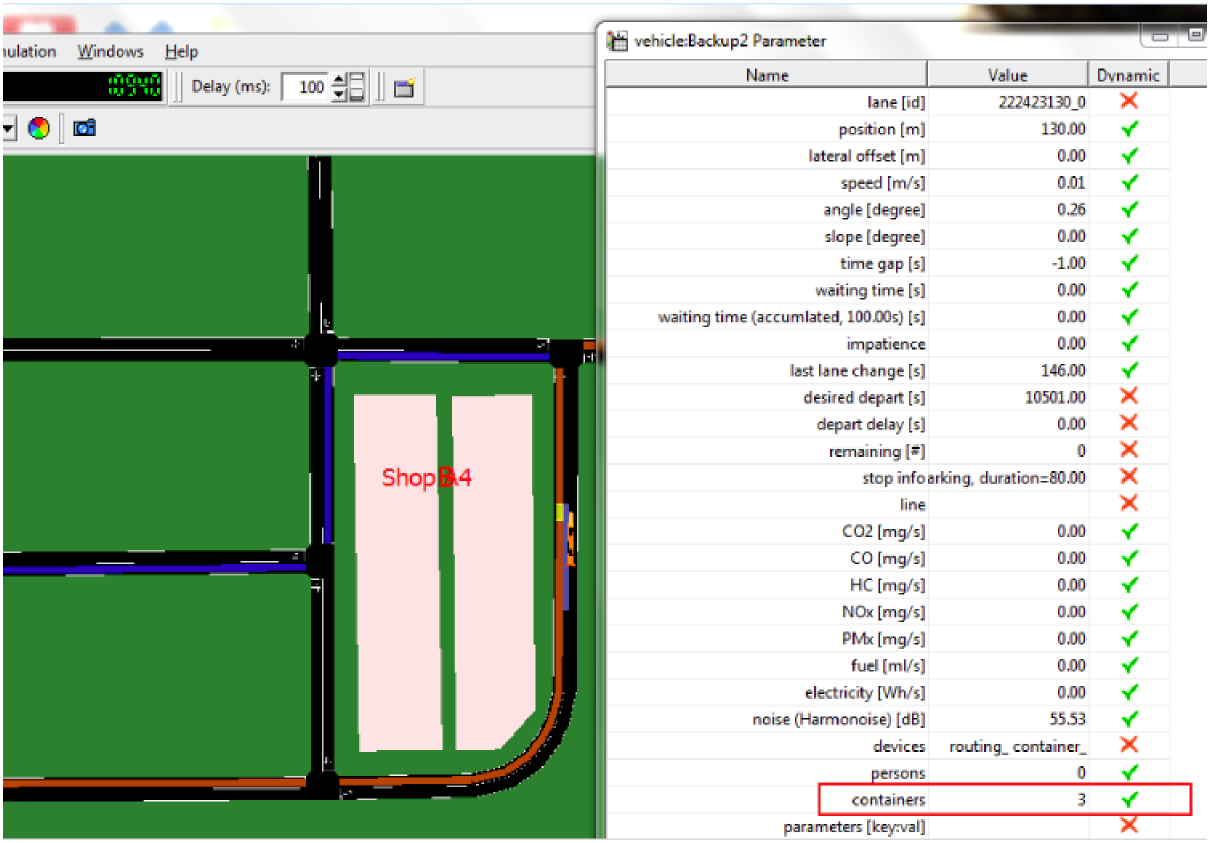


Figure 5.2.3.4: Backup2 Truck Transportation Process at Shop B4

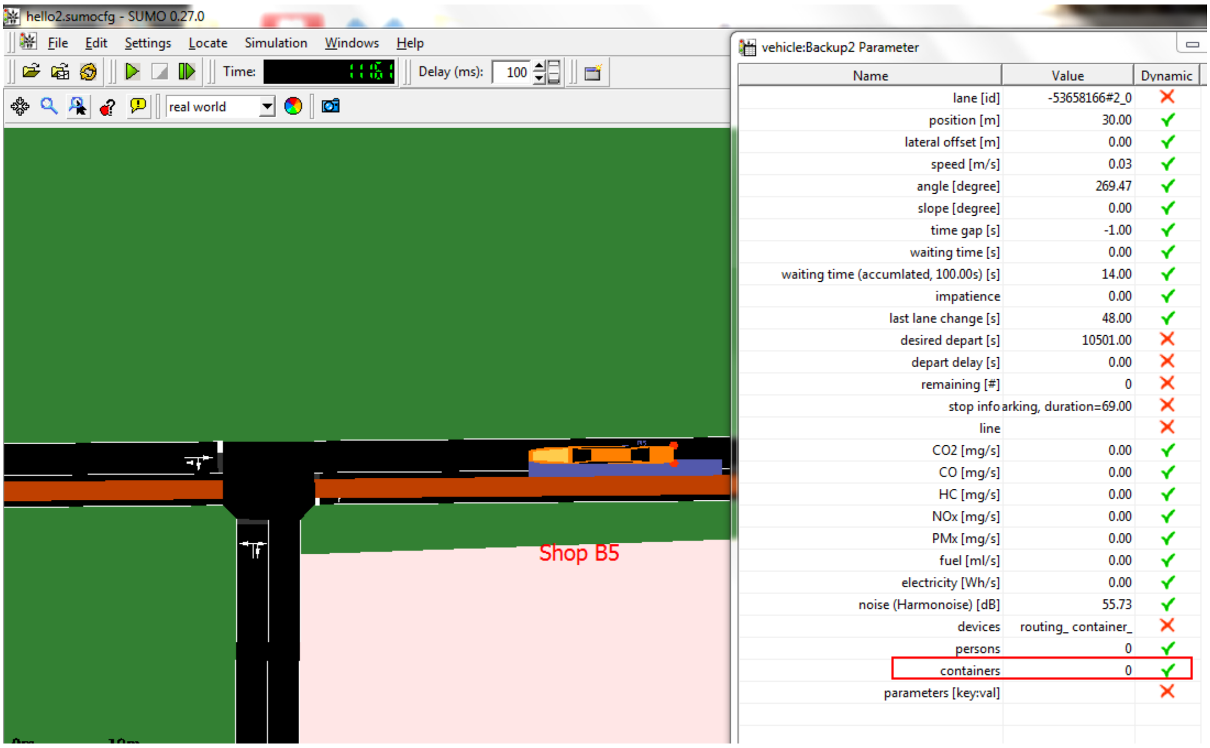


Figure 5.2.3.5: Backup2 Truck Transportation Process at Shop B5

Backup2 Truck starts to reduce the burden of its principal truck which is Truck2 by covering Shop B4 and Shop B5.

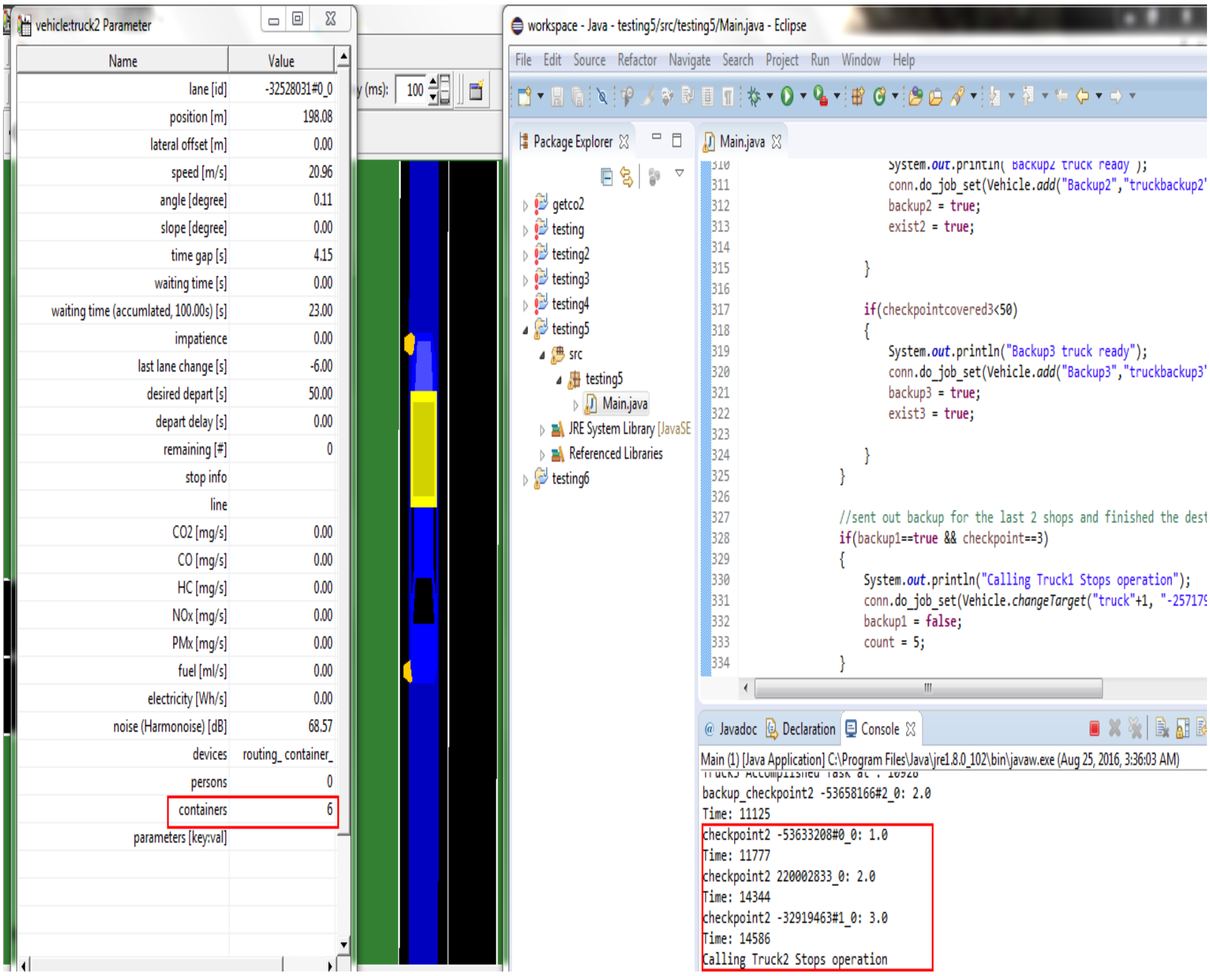


Figure 5.2.3.6: Requiring Truck2 to back to the starting point

Since Backup2 Truck already burden the loads of Truck2 by covering Shop B4 and B5, Truck2 is called back to the starting point after it finished unloading the containers at Shop B1, B2 and B3.

5.3 Concluding Remark

In this chapter, the simulation configurations for automatic rerouting, routing algorithms, logistics and stops of the trucks, the logic flows of backup trucks are discussed. At the same time, the simulation process of truck with or without automatic rerouting capabilities, examples of trucks transportation process and also scenario where backup truck is called upon are illustrated.

CHAPTER 6: SIMULATION EVALUATION AND DISCUSSION

6.1 Simulation Testing and Performance Metrics

In this project, there are 3 criterions that need to be taken into account in order to evaluate the trucks transportation process:

- The petrol consumptions, carbon emissions, time needed for the trucks transportation process during the presence or absence of automatic re-routing capabilities.
- For trucks with automatic re-routing capabilities, the routing algorithms which included Dijkstra's algorithm, A* search algorithm and Contraction Hierarchies will be evaluated based on the petrol consumptions, carbon emissions, time needed for the trucks transportation process
- The time differences to complete the transportation process of the trucks for scenario with backup trucks ready and without the triggering of backup trucks

For each criterions, the lower the petrol consumptions and carbon emissions, as well as the shorter time to complete the transportation process is better.

6.2 Simulation Testing and Result

6.2.1 Result for trucks with the presence or absence of automatic re-routing capabilities

6.2.1.1 Truck without automatic re-routing capabilities

VehicleID	Total Duration (minutes)	Duration complete Transportation Process (minutes)	Route Length (km)	CO (mg/s)	CO^2 (mg/s)	Fuel (ml/s)
Truck1	149.15	114.17	13.92	54085.76	4167485.41	1661.51
Truck2	390.60	368.43	16.81	139270.61	11589817.77	4620.68
Truck3	365.25	357.13	18.45	134450.64	11299163.94	4504.80

Table 6.2.1.1.1: Results of Simulation of Trucks Transportation process without automatic re-routing capabilities

6.2.1.2 Truck with automatic re-routing capabilities

Trucks with automatic re-routing capabilities are evaluated based on the different types of routing algorithms available in SUMO which are Dijkstra's Algorithm, A* search algorithm and Contraction hierarchies

a. Dijkstra's Algorithm

VehicleID	Total Duration (minutes)	Duration complete Transportation Process (minutes)	Reroute No	Route Length (km)	CO (mg/s)	CO^2 (mg/s)	Fuel (ml/s)
Truck1	102.77	58.78	15	26.04	80393.47	6309466.90	2515.49
Truck2	234.32	225.88	43	27.83	121094.35	9590951.62	3823.76
Truck3	204.83	182.00	40	57.00	192180.04	14516996.04	5787.70

Table 6.2.1.1.2: Results of Simulation of Trucks Transportation process with automatic re-routing capabilities using Dijkstra's Algorithm

b. A* search Algorithm

VehicleID	Total Duration (minutes)	Duration complete Transportation Process (minutes)	Reroute No	Route Length (km)	CO (mg/s)	CO2 (mg/s)	Fuel (ml/s)
Truck1	102.77	58.78	15	26.04	80393.47	6309466.90	2515.49
Truck2	234.32	225.88	43	27.83	121094.35	9590951.62	3823.76
Truck3	204.83	182.00	40	57.00	192180.04	14516996.04	5787.70

Table 6.2.1.1.3: Results of Simulation of Trucks Transportation process with automatic re-routing capabilities using A search Algorithm*

c. Contraction Hierarchies Algorithm

VehicleID	Total Duration (minutes)	Duration complete Transportation Process (minutes)	Reroute No	Route Length (km)	CO (mg/s)	CO2 (mg/s)	Fuel (ml/s)
Truck1	148.22	64.73	27	20.02	86180.72	6495955.86	2589.84
Truck2	256.55	250.28	49	32.96	152907.28	11988880.16	4779.78
Truck3	193.75	126.68	38	49.07	171285.29	12615577.73	5029.64

Table 6.2.1.1.4: Results of Simulation of Trucks Transportation process with automatic re-routing capabilities using Contraction Hierarchies Algorithm

6.2.1.3 Results Comparisons and Comments

Trucks that using automatic re-routing capabilities with Dijkstra's Algorithm generally have better statistics comparing to Trucks that using Contraction Hierarchies Algorithm based on results shown in 6.2.1.2 and hence Dikkstra's Algorithm is chosen for the comparisons as follow:

	Total Durations (minutes)		
	Without Automatic Re-routing	**With Automatic Re-routing (Dijkstra's Algorithm)**	**Percentage Changed (%)**
Truck1	149.15	102.77	-31.00
Truck2	390.60	234.32	-40.00
Truck3	365.25	204.83	-43.92
	Duration complete Transportation Process (minutes)		
	Without Automatic Re-routing	**With Automatic Re-routing (Dijkstra's Algorithm)**	**Percentage Changed (%)**
Truck1	114.17	58.78	-48.52
Truck2	368.43	225.88	-38.69
Truck3	357.13	182.00	-49.04
	CO2		
	Without Automatic Re-routing	**With Automatic Re-routing (Dijkstra's Algorithm)**	**Percentage Changed (%)**
Truck1	54085.76	80393.47	+48.64
Truck2	139270.61	121094.35	-13.05
Truck3	134450.64	192180.04	+42.94

	CO (mg/s)		
	Without Automatic Re-routing	**With Automatic Re-routing (Dijkstra's Algorithm)**	**Percentage Changed (%)**
Truck1	4167485.41	6309466.90	+51.40
Truck2	11589817.77	9590951.62	-17.25
Truck3	11299163.94	14516996.04	+28.48
	Fuel (ml/s)		
	Without Automatic Re-routing	**With Automatic Re-routing (Dijkstra's Algorithm)**	**Percentage Changed (%)**
Truck1	1661.51	2515.49	+51.40
Truck2	4620.68	3823.76	-17.25
Truck3	4504.80	5787.70	+28.48

Table 6.2.1.3.1: A summary of the percentage difference for different elements for scenario with and without automatic re-routing capabilities

Based on the table of comparisons above, trucks with automatic re-routing capabilities are gaining huge advantages by at least 31% decreasing total durations and at least 38% decreasing duration required to complete transportation Process comparing to trucks without automatic re-routing capabilities.

However, when comparing to the rate of Carbon, Carbon Dioxide emissions and fuel consumptions, it is noticeable that trucks without automatic re-routing capabilities have better advantages due to the reasons that they are following the pre-determined routes and do not need to re-route for every 300s. Hence, they are having shorter total route length and this results in lower emission rate of Carbon, Carbon Dioxide and lower fuel consumptions comparing to trucks with automatic re-routing capabilities.

6.2.2 Results for Scenario with Backup Trucks

Based on the simulation result, among the 3 trucks, only truck2 is required the help of Backup Truck at timestep of 10500s. The time differences to complete the food transportation process with backup trucks and without backup trucks is as shown:

Duration complete Transportation Process (minutes) in Area B		
Without Backup Truck	**With Backup Truck**	**Percentage Changed (%)**
225.88	219.11	-3.00

Table 6.2.2.1 : Time Comparisons for scenario with Backup Trucks and without Backup Trucks

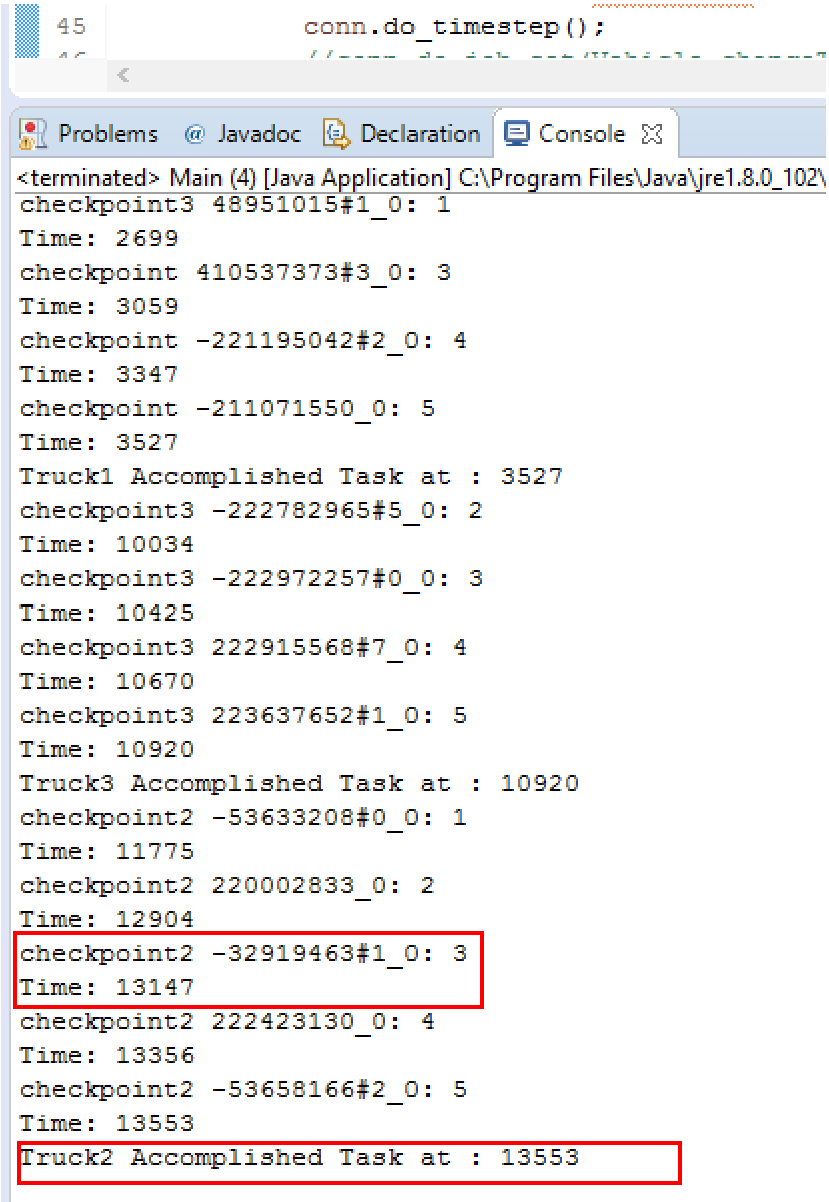


Figure 6.2.2.1: Figure shows the timestep output of Truck2 without the help of backup truck

The figure above shows that at the situation whereby truck2 doesn't receive helps from backup truck, the total time it needs to cover all the 5 shops is 13553s which is 225.88 minutes.

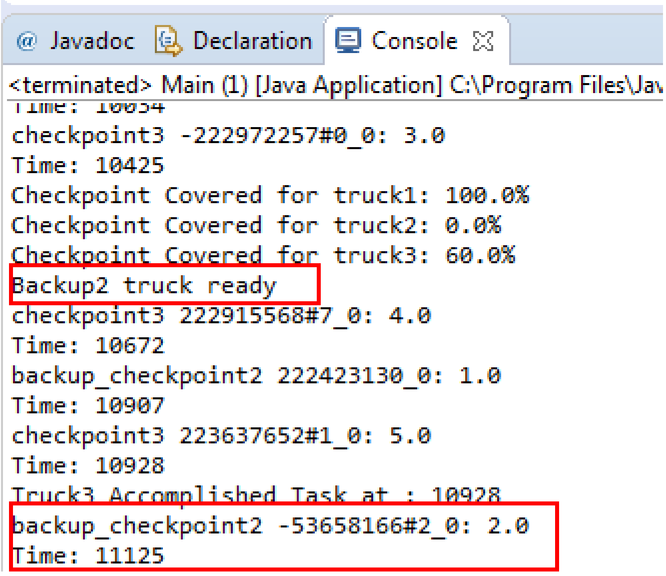


Figure 6.2.2.2: The output of timestep result at the scenario when backup2 truck is called upon

As seen from the figure, in the scenario where truck2 receives help from backup2 truck when reaching timestep of 10500s, the backup2 truck covers the last 2 destinations of truck2 and finished the task at timestep of 11125s. The remaining 3 shops will still have to cover by truck2.

Figure 6.2.2.1 shows that truck2 completed 3 shops at 13147s, since the last 2 shops had been covered by backup2 truck earlier, the transportation process in Area B declared to be completed at timestep of 13147 which is when truck2 completed the 3 shops. Comparing to scenario without the help of backup2 truck, truck2 can only completed the transportation process at timestep of 13553s.

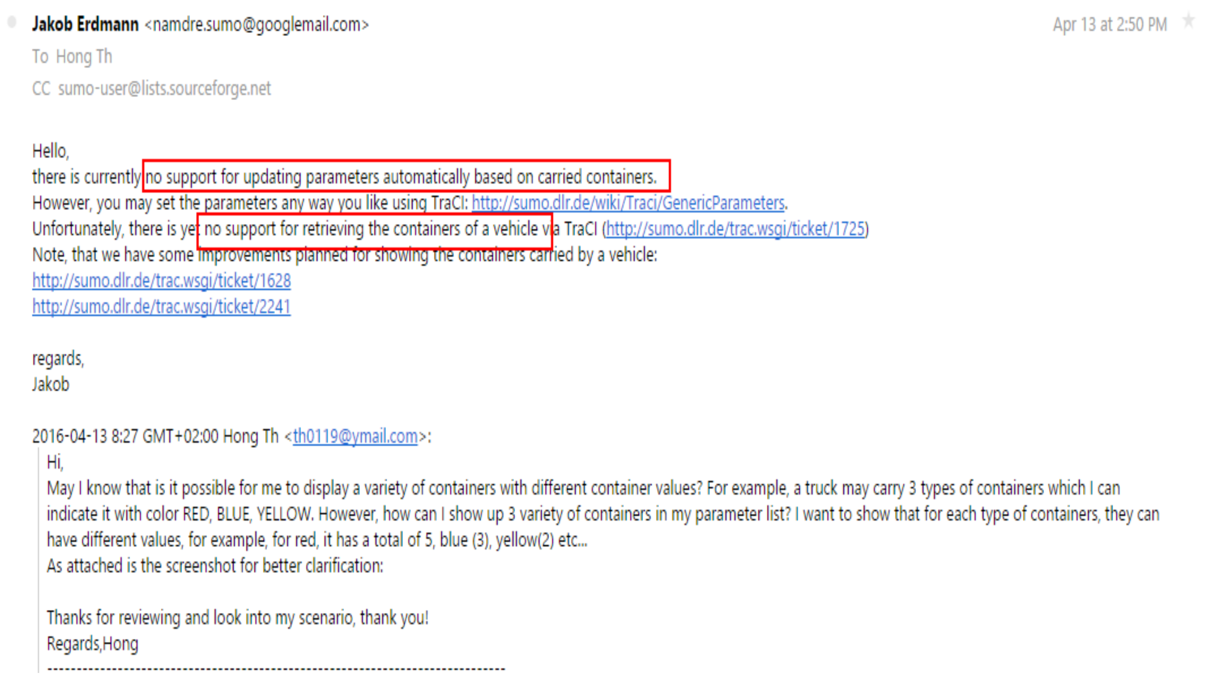
6.3 Project Challenges

Throughout this project, the main challenges are as follow:

a. To simulate the scenario whereby the trucks are able to carry different types of logistics in order to carry them to different locations so that one shop can have various types of logistics to unload and to display the decreasing number of containers when they are unloaded at each shops.

Besides that, the problems also included where user can specify the number of containers required for example 1000 containers without have to hard code the containers into the xml file.

However due to the limitation of SUMO, modelling different types of logistics is still not possible at the current stage. The figure below shows the proof where the email reply from the developers of SUMO explaining the limitations regarding this issue:



Jakob Erdmann <namdre.sumo@googlemail.com> Apr 13 at 2:50 PM

To Hong Th

CC sumo-user@lists.sourceforge.net

Hello,
there is currently no support for updating parameters automatically based on carried containers.
However, you may set the parameters any way you like using TraCI: http://sumo.dlr.de/wiki/Traci/GenericParameters.
Unfortunately, there is yet no support for retrieving the containers of a vehicle via TraCI (http://sumo.dlr.de/trac.wsgi/ticket/1725)
Note, that we have some improvements planned for showing the containers carried by a vehicle:
http://sumo.dlr.de/trac.wsgi/ticket/1628
http://sumo.dlr.de/trac.wsgi/ticket/2241

regards,
Jakob

2016-04-13 8:27 GMT+02:00 Hong Th <th0119@ymail.com>:

Hi,
May I know that is it possible for me to display a variety of containers with different container values? For example, a truck may carry 3 types of containers which I can indicate it with color RED, BLUE, YELLOW. However, how can I show up 3 variety of containers in my parameter list? I want to show that for each type of containers, they can have different values, for example, for red, it has a total of 5, blue (3), yellow(2) etc...
As attached is the screenshot for better clarification:

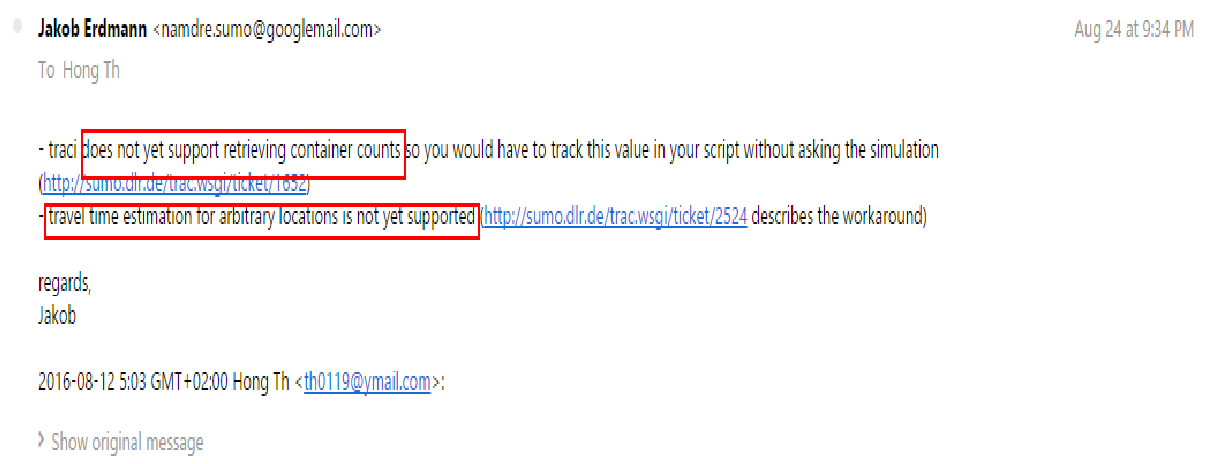
Thanks for reviewing and look into my scenario, thank you!
Regards,Hong

Figure 6.3.1: Email reply from SUMO developer on various types of containers issue

b. The next challenge is about the coordination between trucks so that when one fall into the traffic jam, the other trucks able to help the truck after retrieving the distance between the trucks which fall in traffic, estimated travelling time, container numbers and determined whether or not to assist the truck in traffic.

Again, due to the limitations of SUMO and also Trass, there are no parameters which mentioned here in order to retrieve the container numbers of trucks and estimated travelling time. Following link shows the documentation of Trass: http://traas.sourceforge.net/javadoc/

At the same time, the developer from SUMO confirmed the limitations as mentioned through the email as shown:



Jakob Erdmann <namdre.sumo@googlemail.com> Aug 24 at 9:34 PM

To Hong Th

- traci does not yet support retrieving container counts so you would have to track this value in your script without asking the simulation (http://sumo.dlr.de/trac.wsgi/ticket/1652)
- travel time estimation for arbitrary locations is not yet supported (http://sumo.dlr.de/trac.wsgi/ticket/2524 describes the workaround)

regards,
Jakob

2016-08-12 5:03 GMT+02:00 Hong Th <th0119@ymail.com>:

› Show original message

Figure 6.3.2: Email reply from SUMO developer regarding limitations of retrieval of container numbers and estimated traveling time

Hence, in order to countermeasure this limitation, the best approach to deal with this scenario is by assuming a certain time threshold whereby all the trucks on the map should cover at least 50% of the destined shops. If there are trucks which couldn’t fulfill this requirement, they will have the backup trucks to cover the other half of the shops so as to reduce the burden of the principal trucks. This approach has been discussed in Chapter 5.

6.4 Objectives Evaluation

Following are the objectives which proposed in this project earlier:

- to simulate the traffic situations in cities with food trucks transportation loading and unloading logistics capability by using a traffic simulator
    - Achieved with limitation. By using SUMO simulator, the trucks are able to modelled with containers carried on them and shows the decreasing value when they reach the shops to unload the logistics. However, various types of logistics couldn't be modelled due to the limitation of SUMO as described in Figure 6.3.1. Besides that, the automatic re-routing is able to apply on the trucks as well so as to avoid the traffic jam as much as possible.
- to simulate the transportation process of the food trucks
    - Achieved. Managed to allow trucks to transport the logistics to various destined shops as shown in Chapter 5 section 5.2.1. At the same time, backup truck able to be called when the principal trucks unable to cover at least 50% of the destined shop at certain timesteps.
- to suggest the best routing algorithm for the food trucks to distribute logistics based on the simulation
    - Achieved. In section 6.2.1.2, Dijkstra's Algorithm has been chosen for re-routing due to the better advantage in results comparisons.
- to evaluate the performance of the routing algorithms with following criteria:
    - petrol consumption
    - Carbon emission
    - the time needed to reach the destination to unload the logistics
        - Achieved. In section 6.2.1.2, comparison tables have been constructed so as to determine the value for each elements based on the output trip file generated by the simulation program.

6.5 Concluding Remark

In this section, the simulation testing and results have been discussed and compared based on different scenarios. At the same time, due to the open source nature of SUMO simulator, there are some limitations in it and alternative methods have been proposed to countermeasure the limitations that face. Finally, the objectives in the projects are evaluated as well so as to determine the objectives have been achieved.

CHAPTER 7: CONCLUSION AND RECOMMENDATION

In conclusion, this project has discussed about the issues and problems whereby the unpredictable traffic in the city could slow down the food distribution to the shops. As a consequences, this will bring impacts to the quality of food causing unnecessary wastage of food and the losses to both the distributor and retailers.

Hence, it is necessary to have an intelligent transportation simulation in order to support the food distribution process in the city whereby through the simulation, when there is heavy jam on the road, the truck will reroute so as to get an alternative path which is more time saving rather than the pre-selected path.

In order to do so, SUMO simulator is employed to simulate the city's traffic which included the simulation process of truck transportation to show the summary trip info we can achieve.

Basically, mostly objectives in the projects can be achieved but due to some limitations of SUMO which is an open source software, values retrieval for example, estimated travelling time, containers number of trucks couldn't be achieved, at the same time, modelling of different types of logistics to available as well. Hence, critical thinking and problem solving skills are essential so as to countermeasure the limitation of the software itself.

For further improvement in this project, due to the nature of SUMO which is an open source software, any feedbacks from the users will be logged as a ticket and the developers can further enhance the functionality as well as the features based on the feedbacks and potentially to become a very powerful software to simulate different types of traffic scenarios.

REFERENCES

Aimsum 8.1.3, computer software 2016. Available from: https://www.aimsun.com/. [24 March 2016]

Aragrande, M. and Argenti, O. (2001) *Chapter 1. Food Supply and Distribution Systems to Cities*. Available at: http://www.fao.org/docrep/003/x6996e/x6996e08.htm (Accessed: 16 July 2015).

Arriaga, J., Cárdenas, C., Rivera, J.-P., Sada, J.-C. and Campos, H. (2013) 'A Lean Route to Increase Transportation Agility and Food Freshness', *Industrial and Systems Engineering Research Conference*, , pp. 1–10.

Behrisch, M., Krajzewicz, D. and Weber, M. (eds.) (2014) *Simulation of Urban Mobility*. Springer Berlin Heidelberg.

Design, I. and Ecotrust (2007) *Welcome to FoodHub: Where food people connect.* Available at: https://food-hub.org/ (Accessed: 16 July 2015).

Freidberg, S. (2009) *Fresh: A Perishable History*. Cambridge, MA: Harvard University Press.

Gasson, S. (2005) *Susan Gasson, academic home page*. Available at: http://cci.drexel.edu/faculty/sgasson/SSM/Purpose.html (Accessed: 24 March 2016).

Hu, W., Wang, H. and Yan, L. (2014) 'An actual urban traffic simulation model for predicting and avoiding traffic congestion', *17th International IEEE Conference on Intelligent Transportation Systems (ITSC)*, , pp. 2681–2686. doi: 10.1109/itsc.2014.6958119.

KOTUSEVSKI, G. and HAWICK, K.A. (2009) 'A Review of Traffic Simulation Software', *A Review of Traffic Simulation Software*, 13, pp. 35–54.

Panapinum, K. and Charnsethikul, P. (no date) 'Vehicle Routing and Scheduling Problems', *A Case Study Of Food Distribution In Greater Bangkok*, .

Smith, D., Djahel, S. and Murphy, J. (2014) 'A SUMO based evaluation of road incidents' impact on traffic congestion level in smart cities', *39th Annual IEEE*

Conference on Local Computer Networks Workshops, , pp. 702–710. doi: 10.1109/lcnw.2014.6927724.

Sumo 0.27, computer software 2016. Available from: <http://www.dlr.de/ts/en/desktopdefault.aspx/tabid-9883/16931_read-41000/>. [24 March 2016]

Treiber's Microsimulation of Road Traffic, computer software 2016. Available from: < http://www.traffic-simulation.de/ >. [24 March 2016]

Vigneault, C., Thompson, J., Hui, K.P.C., Wu, S. and I. LeBlanc, D. (2009) 'Transportation of fresh horticultural produce', in Benkeblia, N. (ed.) *Postharvest technologies for horticultural crops 2009*. Trivandrum, Kerala, India: Research Signpost, pp. 1–24.

Villarreal, B. (2012) 'The transportation value stream map (TVSM)', *European J. of Industrial Engineering*, 6(2), p. 216. doi: 10.1504/ejie.2012.045606.

Wakeland, W., Cholette, S. and Venkat, K. (2011) 'Food transportation issues and reducing carbon footprint', in *Green Technologies in Food Production and Processing*. Springer Science + Business Media, pp. 211–236.

Worldatlas (2015) Where Is Subang Jaya, Malaysia?[online]. Available from: http://www.worldatlas.com/as/my/10/where-is-subang-jaya.html (Accessed 25 August 2016)

Zheng, W., Hu, X. and Zeng, A.Z. (2007) 'A Two-Stage Solution Procedure for Food Delivery Decisions in Cities with Circular Transportation Networks', *2007 IEEE International Conference on Service Operations and Logistics, and Informatics*, . doi: 10.1109/soli.2007.4383892.

APPENDIX A: FINAL YEAR PROJECT BIWEEKLY REPORT

Final Year Project Weekly Report

Trimester & Year: Year 3 Semester 3	Study Week No: Week 1 & Week 2
Student Name & ID: Hong Tat Heng (13ACB08246)	
Supervisor: Dr Goh Hock Guan	
Project Title: Intelligent Transportation Simulation for Supporting Smart City Food Distribution	

1. **WORK DONE**
   - Refined Report of FYP1 from Chapter 1 to 2

2. **WORK TO BE DONE**
   - restructure and correct mistakes done in FYP1

3. **PROBLEMS ENCOUNTERED**
   - None

4. **SELF EVALUATION OF THE PROGRESS**
   - Everything is sticked to the schedule

Supervisor's Signature

Student's Signature

Final Year Project Weekly Report

Trimester & Year: Year 3 Semester 3	Study Week No: Week 3 & Week 4
Student Name & ID: Hong Tat Heng (13ACB08246)	
Supervisor: Dr Goh Hock Guan	
Project Title: Intelligent Transportation Simulation for Supporting Smart City Food Distribution	

1. **WORK DONE**
   - Completed Chapter 2 and Chapter 4
   - Using Subang Jaya to restructure the whole simulation previously done in FYP1

2. **WORK TO BE DONE**
   - Implement automatic rerouting on trucks simulation

3. **PROBLEMS ENCOUNTERED**
   - None

4. **SELF EVALUATION OF THE PROGRESS**
   - Everything is sticked to the schedule

Supervisor's Signature

Student's Signature

Final Year Project Weekly Report

Trimester & Year: Year 3 Semester 3	Study Week No: Week 5 & Week 6
Student Name & ID: Hong Tat Heng (13ACB08246)	
Supervisor: Dr Goh Hock Guan	
Project Title: Intelligent Transportation Simulation for Supporting Smart City Food Distribution	

1. **WORK DONE**
   - Completed implementation of automatic re-routing on trucks

2. **WORK TO BE DONE**
   - Consult developer of SUMO for trucks coordination

3. **PROBLEMS ENCOUNTERED**
   - None

4. **SELF EVALUATION OF THE PROGRESS**
   - One of the objectives has been achieved

Supervisor's Signature

Student's Signature

Final Year Project Weekly Report

Trimester & Year: Year 3 Semester 3	**Study Week No: Week 7 & Week 8**
Student Name & ID: Hong Tat Heng (13ACB08246)	
Supervisor: Dr Goh Hock Guan	
Project Title: Intelligent Transportation Simulation for Supporting Smart City Food Distribution	

1. **WORK DONE**
   - Trying alternative ways to perform coordination of trucks while awaiting email reply from SUMO developer

2. **WORK TO BE DONE**
   - Finding ways to implement trucks coordination

3. **PROBLEMS ENCOUNTERED**
   - Limited info to perform trucks coordination

4. **SELF EVALUATION OF THE PROGRESS**

Supervisor's Signature

Student's Signature

Final Year Project Weekly Report

Trimester & Year: Year 3 Semester 3	**Study Week No: Week 9 & Week 10**
Student Name & ID: Hong Tat Heng (13ACB08246)	
Supervisor: Dr Goh Hock Guan	
Project Title: Intelligent Transportation Simulation for Supporting Smart City Food Distribution	

1. WORK DONE

- Completed Chapter 5
- Using Traas to countermeasure limitations of SUMO to perform trucks coordination

2. WORK TO BE DONE

- Complete the rest of the report

3. PROBLEMS ENCOUNTERED

- There's some limitation with SUMO to perform trucks coordination

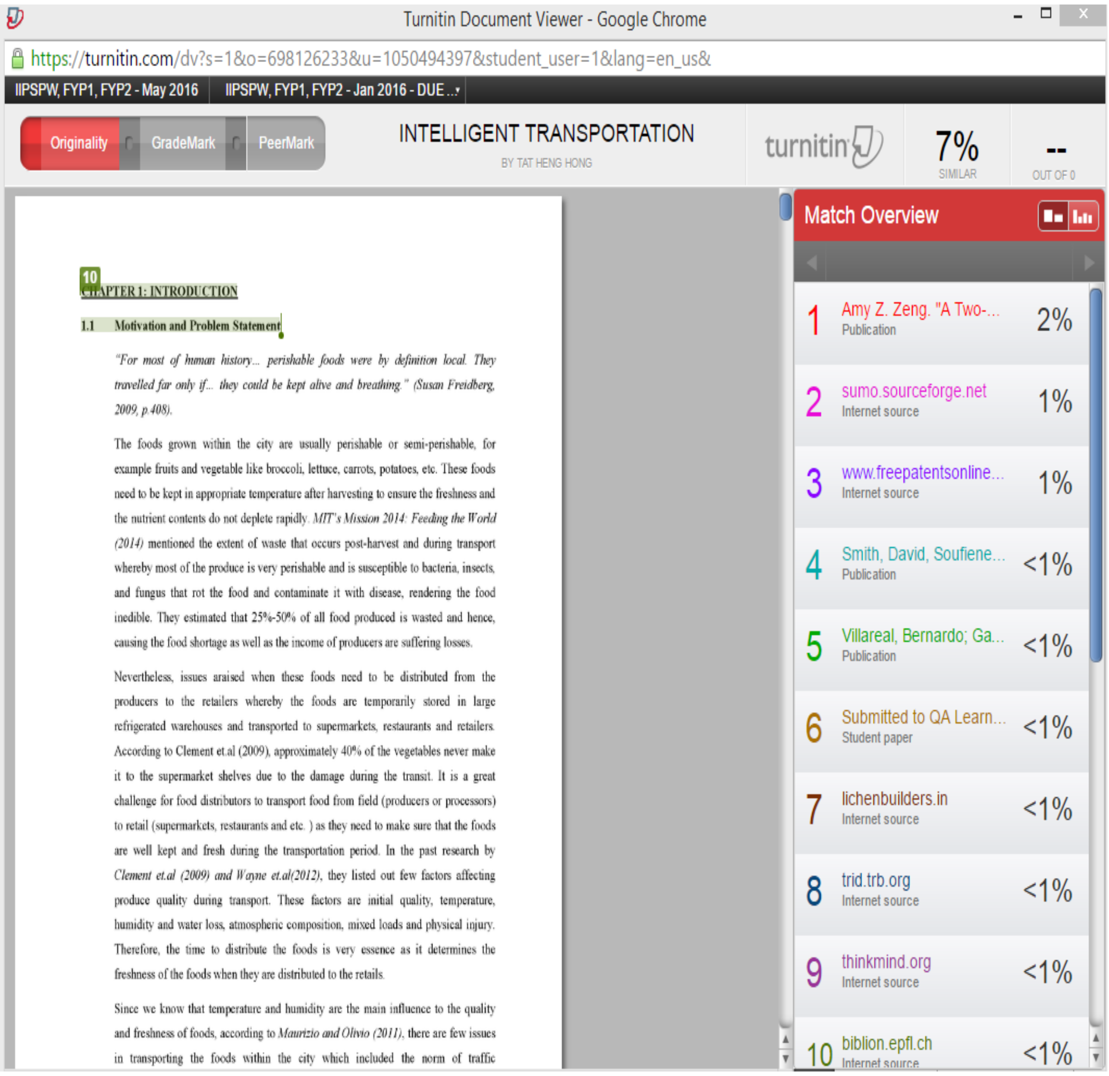
4. SELF EVALUATION OF THE PROGRESS

- Progress still on track

Supervisor's Signature

Student's Signature

APPENDIX B: Plagiarism Check Summary


Turnitin Document Viewer - Google Chrome
https://turnitin.com/dv?s=1&o=698126233&u=1050494397&student_user=1&lang=en_us&
IIPSPW, FYP1, FYP2 - May 2016
IIPSPW, FYP1, FYP2 - Jan 2016 - DUE ...
Originality
GradeMark
PeerMark
INTELLIGENT TRANSPORTATION
BY TAT HENG HONG
turnitin
7%
SIMILAR
--
OUT OF 0
Match Overview
1 Amy Z. Zeng. "A Two-... Publication 2%
2 sumo.sourceforge.net Internet source 1%
3 www.freepatentsonline... Internet source 1%
4 Smith, David, Soufiene... Publication <1%
5 Villareal, Bernardo; Ga... Publication <1%
6 Submitted to QA Learn... Student paper <1%
7 lichenbuilders.in Internet source <1%
8 trid.trb.org Internet source <1%
9 thinkmind.org Internet source <1%
10 biblion.epfl.ch Internet source <1%
10
CHAPTER 1: INTRODUCTION
1.1 Motivation and Problem Statement
"For most of human history... perishable foods were by definition local. They travelled far only if... they could be kept alive and breathing." (Susan Freidberg, 2009, p.408).
The foods grown within the city are usually perishable or semi-perishable, for example fruits and vegetable like broccoli, lettuce, carrots, potatoes, etc. These foods need to be kept in appropriate temperature after harvesting to ensure the freshness and the nutrient contents do not deplete rapidly. MIT's Mission 2014: Feeding the World (2014) mentioned the extent of waste that occurs post-harvest and during transport whereby most of the produce is very perishable and is susceptible to bacteria, insects, and fungus that rot the food and contaminate it with disease, rendering the food inedible. They estimated that 25%-50% of all food produced is wasted and hence, causing the food shortage as well as the income of producers are suffering losses.
Nevertheless, issues araised when these foods need to be distributed from the producers to the retailers whereby the foods are temporarily stored in large refrigerated warehouses and transported to supermarkets, restaurants and retailers. According to Clement et.al (2009), approximately 40% of the vegetables never make it to the supermarket shelves due to the damage during the transit. It is a great challenge for food distributors to transport food from field (producers or processors) to retail (supermarkets, restaurants and etc.) as they need to make sure that the foods are well kept and fresh during the transportation period. In the past research by Clement et.al (2009) and Wayne et.al(2012), they listed out few factors affecting produce quality during transport. These factors are initial quality, temperature, humidity and water loss, atmospheric composition, mixed loads and physical injury. Therefore, the time to distribute the foods is very essence as it determines the freshness of the foods when they are distributed to the retails.
Since we know that temperature and humidity are the main influence to the quality and freshness of foods, according to Maurizio and Olivio (2011), there are few issues in transporting the foods within the city which included the norm of traffic

Universiti Tunku Abdul Rahman			
Form Title : Supervisor's Comments on Originality Report Generated by Turnitin for Submission of Final Year Project Report (for Undergraduate Programmes)			
Form Number: FM-IAD-005	Rev No.: 0	Effective Date: 01/10/2013	Page No.: 1of 1

FACULTY OF INFORMATION AND COMMUNICATION TECHNOLOGY

Full Name(s) of Candidate(s)	Hong Tat Heng
ID Number(s)	13ACB08246
Programme / Course	Communications and Networking
Title of Final Year Project	Intelligent Transportation Simulation for Supporting Smart City Food Distribution

Similarity	**Supervisor's Comments (Compulsory if parameters of originality exceeds the limits approved by UTAR)**
Overall similarity index: 7 % **Similarity by source** Internet Sources: 3 % Publications: 4% Student Papers: 2%	
Number of individual sources listed of more than 3% similarity: 4	
Parameters of originality required and limits approved by UTAR are as follows: **(i) Overall similarity index is 20% and below, and** **(ii) Matching of individual sources listed must be less than 3% each, and** **(iii) Matching texts in continuous block must not exceed 8 words** ***Note: Parameters (i) – (ii) shall exclude quotes, bibliography and text matches which are less than 8 words.***	

Note Supervisor/Candidate(s) is/are required to provide softcopy of full set of the originality report to Faculty/Institute

Based on the above results, I hereby declare that I am satisfied with the originality of the Final Year Project Report submitted by my student(s) as named above.

____
Signature of Supervisor
Name: __________________________

Date: ___________________________

____
Signature of Co-Supervisor
Name: __________________________

Date: ___________________________