

# 博士論文

## TAXI GPS DATA ANALYSIS FOR THE IMPROVEMENT OF TAXI OPERATION OF BANGKOK

(バンコクにおけるタクシー運行改善のための GPS データ分析)

ランジット ソロブ

**TAXI GPS DATA ANALYSIS FOR THE IMPROVEMENT OF  
TAXI OPERATION OF BANGKOK**

**(バンコクにおけるタクシー運行改善のための GPS データ分析)**

A dissertation

by

Saurav Ranjit

In partial fulfillment of the requirements for the degree of  
Doctor of Engineering

Advisor: Associate Professor Yoshihide Sekimoto

Department of Civil Engineering  
The University of Tokyo  
Tokyo, Japan

September 2018

## COMMITTEE MEMBERS

Yoshihide SEKIMOTO, Dr. (Chairperson)  
Associate Professor, Department of Civil Engineering

---

Ryosuke SHIBASAKI, Dr.  
Professor, Center for Spatial Information Science

---

Yukio SADAHIRO, Dr.  
Professor, Center for Spatial Information Science

---

Takashi FUSE, Dr.  
Professor, Department of Civil Engineering

---

Numada MUNEYOSHI, Dr.  
Associate Professor, Department of Civil Engineering

---

## ACKNOWLEDGEMENTS

I wish to take this outstanding opportunity to express my deep gratitude to my respected advisor Professor Ryosuke Shibasaki and Associate Professor Yoshihide Sekimoto at The University of Tokyo, for their exquisite supervision, valuable suggestion and feedback with those never-ending encouragements, all of which have qualified me to accomplish the study.

I would personally like to express my sincere gratitude to respected Associate Professor Masahiko Nagai at Yamaguchi University and Assistant Professor Apichon Witayangkurn at The University of Tokyo for their positive and practical comments on my research work, and in giving extremely valuable recommendations. I am sincerely thankful to respected committee member Professor Sadahiro Yukio, Professor Takashi Fuse and Associate Professor Muneyoshi Numada for providing me with proficient guidance, comments and support to improve my thesis.

I would like to express my gratitude to Toyota Tsusho Nexty Electronics (Thailand) Co., Ltd. by providing Taxi Probe Data for research purpose as well as for the support during period of my research work. It is also my humble pleasure to express reverential thanks to Ms. Sasirin Srisomkiew and Ms. Apantri Peungnumsai for the support during the period of research work.

The work will be incomplete without greeting my beloved mother Mrs. Jagat Devi Ranjit and father Mr. Sri Gopal Ranjit. It is truly because of their continuous support and encouragement in this research work that they have always put their interest in trying to know about the work I am doing. I feel very blessed to have such dynamic parents who have always believed in me and shown me the right path in every moment of my happiness and sorrow.

In addition, I would like to thank all my friends, seniors, junior, associates and colleagues of Shibasaki and Sekimoto lab at The University of Tokyo and Asian Institute of Technology as well as other well-wishers who directly or indirectly shared their companionship with me which I have enjoyed at fullest in my work to come up with the innovative solutions to all the problems that I faced in my work.

## ABSTRACT

With the growing advancement in the field of Global Positioning System (GPS) technology, the utilization of GPS in the field of science has increased significantly in the past recent years. One of the prominent fields which have benefited from the GPS technology is the spatial information science. Spatial information is essentially a digital data that provides information concerning location, people and in times their activities as well. In such, the transportation sector is one of the most benefitted sectors with GPS embedded technology. Whether it is navigation of a vehicle or tracking, GPS technology has been the front-runner for providing the information which further helps decision making. However, technology is not just limited to navigation or tracking. In a recent year, many big cities like New-York and Beijing have started embedding GPS device in the taxi vehicle to collect traffic information. Such vehicle is essentially known as floating car or a probe car. Taxi service are ubiquitous all over the world as a convenient way of commuting in big cities. Bangkok, capital of Thailand, is no exception as more than 100,000 taxis approximately runs daily in and around the city. As, taxi are operational throughout city, mobility data from these vehicles can be an asset for governing urban management and planning. Acknowledging the fact, Toyota Tsusho Nexty Electronics (Thailand) Co. Ltd (TTNET) Bangkok, Thailand has equipped approximately 10,000 GPS devices onto the taxi running in Bangkok city and surrounding provinces. GPS device equipped onto the taxis collects spatial-temporal information every 3-5 seconds with approximately 50 million GPS points per day. Speed, direction and taxi meter status are also collected. Due to the accuracy of GPS, collected data is not always precise and are not always on the road segment. Primarily, GPS data from these probe taxis are utilized for providing the traffic information on a major road segment, however utilization of probe data is not limited to it. The existing literature does shows that there are issues related with the taxi operation in Bangkok, Thailand whether it is from the driver perspective or it is from the passenger perspective. Spatial and temporal data are available from the taxi operation from the Bangkok and surrounding region. These data could be a value asset which could help improve the operation of taxi service through data mining technology. However, lack of proper data infrastructure management system could be the hindrance if proper and efficient mining technology needs to be applied. The detail issue as related is presented as following.

Proper Data Infrastructure Management System: Without proper data infrastructure management system, the data mining working could be a very challenge task especially when dealing with the big data volume. Spatial data involving mobility data from the vehicle movement are constantly increasing. In such cases, how to properly handle the data becomes the primary task before any other data mining algorithm could be any applied.

Taxi Operation Modeling with Quantitative Data Evidence: The issue with the taxi operation services are exist as shown from the past literature. However, model to understand the behavior based on quantitative data evidence are not properly established yet. If the proper behavior model is not established, it could pose a challenge when dealing with the ways to improve the service level of the taxi operations.

Taxi Optimization Modeling: The main two fundamental objectives of the taxi business or the service is to provide good service to the customer or passenger and in turn obtain the monetary profit. However, from the data evidence it is clear that there are issues related with both providing good service as well as getting better monetary profit. Taxi passenger are not happy when they are not provided with the good taxi service or when they are rejected to the service itself. On the other hand, taxi drivers are not getting enough passenger. Though the situation is ironic in nature itself, the problem does exist. One of way to minimize the issue is to optimize the operation of the taxi service. The optimization method as proposed will have ability for the driver in which driver can choose passenger depending upon the passenger origin and destination as well as available demand in the region. Optimization model is to provide recommendation to taxi driver which passenger would be better to choose and which not through mobile application. The hypothesis behind the model is that when driver have ability to choose the passenger then passenger rejection would be drastically minimized as well as choosing passenger would give driver some degree of freedom on how monetary profit could be improved. In addition, optimizing route for efficient taxi operation also plays an important role determining how much profit the driver can make by reducing the operation cost on fuel as well as its maintenance.

The main objective of this research is to help improve taxi operation in Bangkok region through quantitative data analysis from the GPS probe data from taxi. The overall objective is subclass as following

- Develop the data infrastructure management system for big mobility data handling and operation.
- Develop the taxi simulation model of the taxi operation for the Bangkok and the surrounding region
- Develop the optimization model for the improvement of the taxi operation

The objective is designed to address the issue for taxi operation in Bangkok which would provide the taxi driver with assistance that would increase the income, reduce the working hour in turn provide better service level for the passengers.

The research task starts with the development of the data infrastructure. The data infrastructure is further categorized for handling probe data and road network data. Bangkok taxi survey is also conducted for the data preparation. Finally, a data platform namely Horton Data platform is developed to handle the big spatial dataset. The second research task includes taxi simulation model. The simulation model is constructed with multiple variables that are derived from the GPS probe taxi data. The third and the final task includes the optimization of the taxi operation based on proving assistance to the taxi driver to improve the overall taxi operation in the Bangkok and the surrounding regions.

The first part of the research is dedicated to analyzing various map matching techniques that can be utilized efficiently and accurately for big GPS dataset. Road network for map-matching is obtained from the Open Street Map (OSM). OSM road network is cleaned for topological error like floating links, pseudo nodes using spatial operation on PostgreSQL. Total of 1,107,798 road link feature is extracted for Thailand which is converted to Well Known Text (WKT) for using in map matching process. Map matching is performed considering road geometry, topology, and connectivity. The simplest geometry operation for matching operation is buffer operation which finds line segment from GPS point within buffer distance. However, identifying optimum buffer

distance is difficult as point accuracy and complexity of the road link is different at various locations. Hausdorff distance matrix is applied for topological operation by taking consecutive 4 GPS points. The Hausdorff distance matrix measures degree of similarity between road link geometry with the line segment geometry created from 4 consecutive GPS point. Topology operation improved the accuracy, but its accuracy could be compromised near the road intersection where consecutive GPS point lies between different road segments. Finally, probabilistic approach in which both geometry, topology is considered along with new parameter of road link connectivity. The probabilistic approach is performed using Hidden Markov Model with Hausdorff distance matrix. Distance matrix provided road link candidate of GPS point for which initial probability, measurement probability and forward transition probability is measured, and GPS point is matched with road link. Accuracy assessment for all the map matching technique is performed for various cases such as ‘single and multiple lane road’, ‘road intersection’ etc. As mentioned, both data as well as road link are very huge, computation time is a critical factor for efficient every operation. A distributed computing platform for large scale data which utilized Hadoop/Hive is used for all computation involved in map-matching operation. The result obtained is labelled GPS data, based on road network, which is utilized for accurately map traffic congestion, improve taxi operations through optimizing taxi routes. By using OSM road networks, the techniques could be replicated and implemented for other country where OSM is available.

The second part of the research is dedicated to model the taxi operation in the Bangkok. Taxi behavior is a spatial–temporal dynamic process involving discrete time dependent events, such as customer pick-up, customer drop-off, cruising, and parking. Simulation models, which are a simplification of a real-world system, can help understand the effects of change of such dynamic behavior. An agent-based modeling and simulation is utilized, that describes the dynamic action of an agent, i.e., taxi, governed by behavior rules and properties, which emulate the taxi behavior. Taxi behavior simulations are fundamentally done for optimizing the service level for both taxi drivers as well as passengers. Moreover, simulation techniques, as such, could be applied to another field of application as well, where obtaining real raw data are somewhat difficult due to privacy issues, such as human mobility data or call detail record data. This research describes the development of an agent-based simulation model which is based on multiple input parameters (taxi stay point cluster; trip information (origin and destination); taxi demand information; free taxi



movement; and network travel time) that were derived from taxi probe GPS data. As such, agent's parameters were mapped into grid network, and the road network, for which the grid network was used as a base for query/search/retrieval of taxi agent's parameters, while the actual movement of taxi agents was on the road network with routing and interpolation. The results obtained from the simulated taxi agent data and real taxi data showed a significant level of similarity of different taxi behavior, such as trip generation; trip time; trip distance as well as trip occupancy, based on its distribution. As for efficient data handling, a distributed computing platform for large-scale data was used for extracting taxi agent parameter from the probe data by utilizing both spatial and non-spatial indexing technique.

The last and final part of the research work focuses on the optimization of the taxi behavior model. For the case of optimization scenario is considered i.e. how the driver pick up the passenger. In this scenario, driver have the ability to choose which passenger to pick and which not to pick. The reason behind of having driver choose the passenger is that the refusal rate would decrease drastically. This does not guarantee that all taxi driver is willing to server the customer. However, if certain number of taxi are not willing to server the customer, there will exist other group of vacant taxi driver that are willing to serve the customer making the demand and supply theoretically in the state of equilibrium unless demand out run the supply.

## TABLE OF CONTENTS

CHAPTER	TITLE	PAGE
	Title Page	i
	Committee Members	ii
	Acknowledgements	iii
	Abstract	iv
	Table of Contents	ix
	List of Figures	xiii
	List of Tables	xvii
	List of Abbreviations	xviii
	List of Notations	xx
<b>1</b>	<b>INTRODUCTION</b>	
	1.1 Background	1
	1.2 Related Work	3
	1.3 Problem Statement	8
	1.4 Research Objective	10
	1.5 Key contribution	10
	1.6 Research main tasks	13
	1.7 Scope and limitation	14
	1.8 Structure of the thesis	14
<b>2</b>	<b>DATA INFRASTRUCTURE</b>	
	2.1 Data Infrastructure Platform	16
	2.1.1 Data Infrastructure Development	16
	2.1.2 Big Data	17
	2.1.3 Hadoop Distributed System	17
	2.1.4 Horton Data Platform	18

2.2	Probe Vehicle Data	20
2.3	Road Network Data	22
2.4	Preliminary Data Analysis	25
2.5	Map Matching	26
2.5.1	Buffer Operation	28
2.5.2	Hausdroff Distance Similarity	29
2.5.3	Probabilities Map Matching	30
2.6	Map Matching Accuracy Assessment Data Preparation	35
2.7	Map Matching Evaluation	39
2.8	Map Matching Test Case Accuracy Evaluation	42
2.9	Map Matching Variable Sampling Rate Accuracy Evaluation	43
2.10	Map Matching Speed Test Performance	44
2.11	Bangkok Taxi Survey	45
<b>3</b>	<b>TAXI BEHAVIOUR SIMULATION</b>	
3.1	Introduction	47
3.2	Taxi Behavior Modeling State	48
3.3	System Overview	52
3.4	Road Network and Grid Network	54
3.5	Data Preparation	56
3.5.1	Data Aggregation	56
3.5.2	Stay Point Extraction	57
3.5.3	Vacant Taxi Movement	61
3.5.4	Taxi Origin and Destination	63
3.5.5	Taxi Demand	70
3.5.6	Network Travel Time	72
3.6	Agent Based Model	75
3.6.1	Initialization Module	76
3.6.2	Taxi Agent	77
3.6.3	Passenger Pick Up Module	78

3.6.4	Passenger Drop Off Module	79
3.6.5	Data Logger and Updater Module	81
3.7	Model Evaluation	82
3.7.1	Distribution Overlapping Coefficient	83
3.7.2	Average Speed Hourly Variation	89
3.7.3	Taxi Occupancy Evaluation	91
3.7.4	Taxi Fare Evaluation	91
3.8	Model Improvement	94
3.8.1	Demand Update	95
3.8.2	Distributed System Implementation	97
<b>4</b>	<b>OPTIMIZATION OF TAXI OPERATION</b>	
4.1	Introduction	99
4.2	Optimization Policy	101
4.3	Passenger Search Policy	102
4.4	Model Evaluation	104
4.4.1	Passenger Waiting Time Evaluation	104
4.4.2	Distance Travel Without Passenger Evaluation	106
4.4.3	Driver's Income Evaluation	107
4.4.4	Passenger Service Level Evaluation	108
<b>5</b>	<b>CONCLUSION</b>	
5.1	Data Infrastructure	111
5.2	Taxi Simulation Modeling	112
5.3	Taxi Optimization Modeling	113
	APPENDIX A	115
	APPENDIX B	120
	APPENDIX C	125
	APPENDIX D	141

APPENDIX E	151
APPENDIX F	157
REFERENCES	167

## LIST OF FIGURES

<b>FIGURE</b>	<b>TITLE</b>	<b>PAGE</b>
Figure 1.1:	Taxi operation issues in Bangkok .....	8
Figure 1.2:	Research contribution on data infrastructure.....	11
Figure 1.3:	Research contribution on simulation modeling.....	12
Figure 1.4:	Research main tasks .....	13
Figure 2.1:	Need for data infrastructure.....	16
Figure 2.2:	MapReduce architecture of hadoop system.....	18
Figure 2.3:	Horton data platform interface .....	19
Figure 2.4:	Probe vehicle working.....	20
Figure 2.5:	Probe data example .....	21
Figure 2.6:	Probe data error example.....	22
Figure 2.7:	Open street map topological error .....	22
Figure 2.8:	Detailed flow diagram of topological error correction.....	24
Figure 2.9:	Open street map (OSM) Thailand road network .....	25
Figure 2.10:	Road link candidate selection at different buffer distance .....	28
Figure 2.11:	Sequence of GPS points for hausdroff distance analysis .....	29
Figure 2.12:	Probabilistic map matching term definition .....	31
Figure 2.13:	Flow diagram of map matching with probabilistic approach.....	33
Figure 2.14:	Selected GPS data points for accuracy assessment.....	35
Figure 2.15:	Different test cases including road type and GPS data type.....	36
Figure 2.16:	Ground truth data preparation with test case.....	37
Figure 2.17:	Map matching accuracy with hausdroff distance approach .....	39
Figure 2.18:	Map matching accuracy with buffer distance approach.....	40
Figure 2.19:	Map matching accuracy with probabilistic approach.....	41
Figure 2.20:	Map matching accuracy evaluation for seven test cases .....	42

Figure 2.21: Map matching accuracy with varying sampling interval .....	43
Figure 2.22: Speed performance test of map matching .....	44
Figure 2.23: Bangkok taxi questionnaire survey .....	46
Figure 3.1: Need for taxi behavior simulation .....	47
Figure 3.2: Taxi behavior modeling state .....	48
Figure 3.3: Conceptual design of agent-based modeling.....	51
Figure 3.4: System overview .....	53
Figure 3.5: Left Open street map of Bangkok road network; Right Grid network.....	55
Figure 3.6: Probe data aggregation in OSM road network and grid network.....	56
Figure 3.7: Flow diagram of a data aggregation process .....	56
Figure 3.8: Data aggregation example .....	57
Figure 3.9: Stay point extraction.....	58
Figure 3.10: Stay point cluster with DBSCAN algorithm .....	59
Figure 3.11: Example for the error cluster detected with DBSCAN .....	59
Figure 3.12: Stay point cluster with grid DBSCAN algorithm.....	60
Figure 3.13: Left: Stay point cluster; Right: Kernel density function of timestamp. ....	61
Figure 3.14: Nine cardinal direction for vacant taxi movement .....	62
Figure 3.15: Direction probability density of in a grid at given time interval .....	62
Figure 3.16: Passenger trip based on pick-up and drop-off transition.....	63
Figure 3.17: Unusual high number of trip in certain taxi .....	64
Figure 3.18: Number of taxi trip according to taxi questionnaire survey.....	65
Figure 3.19: Frequent transition between pickup and drop off.....	65
Figure 3.20: Trip distribution considering trip more than 1 Km .....	66
Figure 3.21: (A) OSM road network and grid network OD comparison; (B) Pick up location at origin with respect to OSM road network; (C) Drop off location at destination with respect to OSM road network .....	68

Figure 3.22: Passenger trip OD at time interval. Left: 7-8 a.m.; Right: 18-19 p.m. ....	69
Figure 3.23: Aggregated taxi demand at time interval. Top:7-8 a.m.; Bottom 18-19 p.m. ....	71
Figure 3.24: Average speed profile on road network segment at different time interval .....	74
Figure 3.25: Agent-based simulation model .....	75
Figure 3.26: Initialization module.....	76
Figure 3.27: Passenger pick up module .....	78
Figure 3.28: Taxi passenger pick up policy .....	79
Figure 3.29: Passenger drop off module .....	80
Figure 3.30: Taxi passenger drop off policy.....	80
Figure 3.31: Data logger and updater module .....	81
Figure 3.32: Weekdays trip distribution comparison.....	84
Figure 3.33: Weekdays trip per grid distribution comparison .....	84
Figure 3.34: Weekdays trip time distribution comparison.....	85
Figure 3.35: Weekdays trip distance distribution .....	85
Figure 3.36: Weekends trip distribution comparison.....	87
Figure 3.37: Weekends trip per grid distribution.....	87
Figure 3.38: Weekends trip time distribution .....	88
Figure 3.39: Weekends trip distance distribution .....	88
Figure 3.40: Average speed variation concerning hourly time interval.....	89
Figure 3.41: Simulated taxi agent trajectory visualization for weekdays simulation .....	90
Figure 3.42: Taxi occupancy. Left: Simulated taxi agent data; Right Real taxi data .....	91
Figure 3.43: Taxi income distribution for weekday and weekend.....	93
Figure 3.44: Passenger updated pick up module .....	95
Figure 3.45: Trip distribution for the updated pick up module (day type = weekdays) .....	97
Figure 3.46: Taxi agent distributed across Bangkok and surrounding provinces.....	98
Figure 4.1: Taxi operation optimization .....	100



Figure 4.2: Optimization or improvement choice.....	101
Figure 4.3: Passenger Search Policy.....	103
Figure 4.4: Passenger waiting time comparison (3000 Taxi Agents).....	105
Figure 4.5: Passenger waiting time comparison (5000 Taxi Agents).....	105
Figure 4.6: Passenger waiting time comparison (10000 Taxi Agents).....	105
Figure 4.7: Daily distance travel without passenger comparison (3000 Taxi Agents).....	106
Figure 4.8: Daily distance travel without passenger comparison (5000 Taxi Agents).....	106
Figure 4.9: Daily distance travel without passenger comparison (10000 Taxi Agents).....	107
Figure 4.10: Driver’s daily income comparison (3000 Taxi Agents).....	107
Figure 4.11: Driver’s daily income comparison (5000 Taxi Agents).....	108
Figure 4.12: Driver daily income comparison (10000 Taxi Agents).....	108
Figure 4.13: Taxi passenger trip per day comparison (3000 Taxi Agents) .....	109
Figure 4.14: Taxi passenger trip per day comparison (5000 Taxi Agents) .....	109
Figure 4.15: Taxi passenger trip per day comparison (10000 Taxi Agents) .....	109
Figure 5.1: Advantage of data infrastructure .....	111

## LIST OF TABLES

<b>TABLE TITLE</b>	<b>PAGE</b>
Table 2.1: Probe data specification.....	21
Table 2.2: Probabilistic map matching example.....	34
Table 2.3: Sample ground truth data.....	38
Table 2.4: Distribution of GPS points with different test cases.....	38
Table 3.1: Passenger and non-passenger trip.....	67
Table 3.2: Agent data based on stay point cluster .....	77
Table 3.3: Weekday simulated trip data vs real trip data comparison .....	83
Table 3.4: Weekend simulation trip data vs real trip data comparison.....	86
Table 3.5: Sample simulated taxi data .....	90
Table 3.6: Taxi fare structure in Bangkok, Thailand.....	92
Table 3.7: Driver income distribution.....	94

## LIST OF ABBREVIATIONS

ABM	Agent-Based Modeling
ABMS	Agent-Based Modeling and Simulation
AI	Artificial Intelligence
CLI	Command Line Interface
DBSCAN	Density Based Spatial Clustering of Application with Noise
GMM	Gaussian Mixture Model
GPS	Global Positioning System
HBSTR-Tree	Hybrid Spatial Temporal R-Tree
HDBSCAN	Hierarchical Density Based Spatial Clustering of Application with Noise
HDFS	Hadoop Distributed File System
HDP	Horton Data Platform
HMM	Hidden Markov Model
HQL	Hive Query Language
IMEI	International Mobile Equipment Identification
ITS	Intelligent Transportation Systems
IVMM	Interactive Voting-based Map Matching
JTS	Java Topological Suite
LAMM	look ahead map matching
MR	Map Reduce
NTP	Network Time Protocol
OD	Origin Destination
OSM	Open Street Map
PBF	Protocolbuffer Binary Format
PSL	Path Size Logit
R-Tree	Rectangle-Tree
SQL	Structured Query Language
SR-Tree	Sphere Rectangle-Tree
ST	Spatial Temporal
STR-Tree	Sort Tile Recursive-Tree

TTNET	Toyota Tsusho Nexty Electronics (Thailand) Co., LTD.
UDAF	User Defined Aggregated Function
UDF	User Defined Function
VSW	Variable Sliding Window
WKT	Well Known Text
YARN	Yet Another Resource Negotiator

## LIST OF NOTATIONS

$\langle \text{key}, \text{value} \rangle$	Key Value pair
$H(A, B)$	Hausdorff distance similarity index
$A = \{a_1, a_2.. a_n\}$	Finite sets of GPS points belonging to set A
$B = \{b_1, b_2.. b_n\}$	Finite sets of GPS points belonging to set B
$D(a, b)$	Distance between finite set of GPS points belonging to set A and set B
$Max$	Maximum function
$Min$	Minimum function
$a \in A$	Entity a belongs to set A
$b \in B$	Entity b belongs to set B
$r_i, r_j$	Candidate road link segment
$Z_t$	Probe GPS point at timestamp t
$Z_{t+1}$	Probe GPS point at timestamp t+1
$X_{t,i}$	Projected coordinate at timestamp t
$X_{t+1,i}$	Projected coordinate at timestamp t+1
$P(Z_t \rightarrow r_i)$	Initial probability
$ Z_t \rightarrow r_i $	Euclidean distance between the GPS point $Z_t$ and candidate road link segment $r_i$
$P(Z_t r_i)$	Measurement probability
$ Z_t \rightarrow X_{t,i} $	Great circle distance between the observed GPS point $Z_t$ and projected coordinate point on the road link segment $r_i$
$exp$	Exponential function
$\sigma_z$	Standard deviation of the GPS measurement
$P(X_t \rightarrow X_{t+1})$	Transition probability
$ Z_t \rightarrow Z_{t+1} $	Great circle distance between two consecutive GPS point i.e. $Z_t$ and $Z_{t+1}$
$ X_{t,i} \rightarrow X_{t+1,j} $	Route distance between the projected coordinates of the observed GPS points $Z_t$ and $Z_{t+1}$ on to the road link segment $r_i$ and $r_j$ respectively
$\beta$	Probability parameter
$\pi$	Pi (Value = 3.1415...)

$P_{max}$	Maximum probability
$T_i$	Spatial trajectory generated by moving taxi in geographical space
$P_j$	Coordinate of the moving taxi at a given timestamp
$x_j$	Longitude coordinate of the moving taxi
$y_j$	Latitude coordinate of the moving taxi
$t_j$	Timestamp of the moving taxi
$R$	OSM road network
$R$	Agent rule
$R$	Correlation coefficient
$G$	Grid network of size 500x500 meter
$I$	Agent input
$S$	Agent state
$\Delta t$	Time step chosen for probe data aggregation
$P_{anchor}$	Anchor point for stay point extraction
$P_{successor}$	Successor points for stay point extraction
$D_{threshold}$	Distance threshold for stay point extraction
$T_{threshold}$	Time threshold for stay point extraction
$MinPts$	Minimum points for DBSCAN clustering algorithm
$\forall$	For all entity
$\forall_g \in G_t$	For all the grid $g$ that belongs to grid network $G$ at time interval $t$
$\forall_r \in R_t$	For all the road segment $r$ that belongs to the road network $R$ at time interval $t$
$d$	Nine cardinal direction for the vacant taxi movement
$\forall_g \in G_t, P(d)_g$	Direction probability for the vacant taxi movement, moving to direction $d$ , for all grid $g \in G$ at time interval $t$
$n_g$	Number of vacant taxi points moving to direction $d$ at time interval of $t$
$N_g$	Total number of vacant taxi points in grid $g \in G$ at time interval of $t$
$O$	Origin grid for taxi trip
$D$	Destination grid for taxi trip

$\forall g \in G_t, P(g_{O \rightarrow D})$	Origin Destination probability for all grids $g$ that belongs to $G$ at time interval $t$
$Trip_{O \rightarrow D}$	The total number of passenger trips between the origin grid $O$ and the destination grid $D$
$Trip_O$	All the passenger trips that originated at grid $O$ at time interval $t$
$\forall g \in G_t, P(dm)_g$	Probability of success for all grid $g \in G$ at time interval $t$
$O_g$	Total number of taxi demands generated at grid $g \in G$ and time interval $t$
$V_g$	Total number of recorded vacant taxis at grid $g \in G$ and time interval $t$
$\forall r \in R_t, \bar{s}_r$	Average speed on the road network segment $r \in R$ at a time interval of $t$
$\sum S_{p \in r}$	Sum of the speed of all the points $p$ on the road network segment $r \in R$ at a time interval of $t$
$N_{p \in r}$	Total number of points on the road network segment $r \in R$ at a time interval of $t$
$\forall g \in G_t, \bar{s}_g$	Average speed on the grid network $g \in G$ at a time interval of $t$
$\sum S_{p \in G}$	Sum of the speed of all the points $p$ on the grid network $g \in G$ at a time interval of $t$
$N_{p \in G}$	Total number of points on the grid network $g \in G$ at a time interval of $t$
$\hat{p}$	Estimated interpolated GPS points on the road network segment $r \in R$ at a time interval of $t$
$T_{r: \hat{p} \in t}$	Road network travel time with $r_{distance}$ as road network segment distance
$r_{distance}$	Road network segment distance
$\forall g \in G_t, Pr(dm)_g$	Improved probability of success for all grid $g \in G$ at time interval $t$
<i>Pick up distance</i>	Distance from the taxi agent to the passenger origin
<i>Trip distance</i>	Passenger trip distance from the passenger origin to the passenger destination

# CHAPTER 1

## INTRODUCTION

1.

### 1.1 Background

With the growing advancement in the field of Global Positioning System (GPS) technology, the utilization of GPS in the field of science has increased significantly in the past recent years. One of the prominent fields which have benefited from the GPS technology is the spatial information science. Spatial information is essentially a digital data that provides information concerning location, people and in times their activities as well. In such, the transportation sector is one of the most benefitted sectors with GPS embedded technology. Whether it is navigation of a vehicle or tracking, GPS technology has been the front-runner for providing the information which further helps decision making. However, technology is not just limited to navigation or tracking. In a recent year, many big cities like New-York and Beijing have started embedding GPS device in the taxi vehicle to collect traffic information (N. J. Yuan et al. 2013). Such vehicle is essentially known as floating car or a probe car. Taxi service is ubiquitous all over the world as a fast and convenient way of commuting in big cities. Bangkok, capital of Thailand, is no exception as more than 100,000 taxis approximately run daily in and around the city. Acknowledging the fact, Toyota Tsusho Nexty Electronics (Thailand) Co., Ltd, Bangkok, Thailand has equipped approximately 10,000 GPS devices onto the taxi running in Bangkok city and surrounding provinces which collects data at the sampling rate of 3 or 5 seconds. With such high sampling data, the average volume of data point collected for the probe vehicle could exceed more than 40 million each day easily. Mobility data as such location and time from these vehicles with GPS embedded can be an asset for governing urban management and planning, however collection of data from the massive fleet of the vehicle could significantly create a bottleneck for handling such big data. A proper data management platform is required to handle the massive quantity of dataset. In this research, a cloud-based data management platform namely Hadoop Distributed System (Witayangkurn et al. 2012) and Horton Data Platform (HDP) is utilized to handle big dataset. Spatial data processing, Apache Hive based query HiveQL (Hive Query Language) developed including Hive UDF (User Defined Function) and Hive UDAF (User Defined Aggregated Function) is prepared. The



distributed computing platform is not only for large-scale support but also for fast processing, spatial support and scalable regarding both processing speed and storage (Witayangkurn et al. 2012).

Data obtained from the GPS probe vehicle cannot always guarantee its accuracy. The reason could relate to GPS measurement device itself or due to the external environmental factors such as signal obstruction from the high-rise building, multipath error or even ionospheric error etc. Error in general could simple be the systematic error or the random error (Z. Zheng et al. 2014). Cleaning of the erroneous data is an up most important preliminary task before further processing can be made. In this research, multiple map matching operation with the road network provided by Open Street Map (OSM), is compared together out of which the map matching operation with highest accuracy is chosen. As similar, road network data from OSM is also affected by the its overall accuracy. The network data is cleaned for error with performing network assessment and spatial query and finally topology is validated. As the dataset is large in volume and size the distributed computing platform comes in great usefulness when working with cleaning operation.

Talking about taxi services in Bangkok, as mentioned there are more than 100,000 registered taxi running in and around the Bangkok, Thailand and the surrounding provinces (Peungnumesai et al. 2017). However, there are many issues related with the taxi services. The issue is primary associated with driver's perspective, passenger's perspective as well as environmental perspective. Regarding the issue from driver's perspective, taxi drivers are working longer hour, however the income generated does not justifies their working hour i.e. work long hour for less income which suggest driver are not getting enough passenger. Also, taxi drivers are subjected for high price for the fuel and gas. On the contrary, passenger's perspective for the existing issue with the taxi service is that, passengers are constantly rejected or denied for the service they are entitled to (S. Zhang and Wang 2016). Moreover, taxi drivers often fix the fare charge to go to the passenger's destination place. And finally, with as the environmental perspective, taxi driver tends to work with the intuition, caring less for the proper routing. If proper routing is not taken it could easily cause taxi drivers to take more congested road network to go to the passenger destination. Getting stuck on the traffic congestion means higher co2 emission without any productive means which directly or indirectly could affect the environment. The optimization of the driver behavior model

could be one of the main aspect that could help address the issue. However, to make an optimization, a micro level simulation of the driver behavior is required based on the quantitative data evidence. In this research, an agent-based simulation and modeling is utilized to model the taxi operation in the Bangkok and surrounding provinces. Agent based modeling, which works on the state-rule-input architecture (Torrens 2010), each taxi behaves as an agent, interact with the environment to capture dynamic behavior through reconstructing complex patterns by defined of behavior rules (Baster et al. 2013). As such, agent described by spatial and temporal parameter interacts with an environment which in turn provides the sets of behavior rule that directs the outcome or a goal of the simulation.

## 1.2 Related Work

With the rapid increase of spatial data everyday it has become more and more important for high-speed and efficient query on the data. Spatial index, is a data structure that refers to the location of spatial objects. Spatial index is the vital technique for improving storage efficiency and affecting spatial retrieve performance of spatial data for which many structure and technical of spatial index have been proposed and implemented (Wei et al. 2013). Indexing structure SR- tree is based on R-tree and SS-tree with corresponds to the nested hierarchy of region with utilization of bounding spheres and bounding rectangles (Katayama & Satoh, 1997). (Francis et al. 2008) Proposed indexing structure Q-hash that utilizes the concept of a region based Quadtree which is mostly successful in handling the overlap queries. Moreover, HBSTR-tree supports real-time incremental trajectory databases in which spatio-temporal R-tree is the principal part, which supports spatio-temporal range query with Hash table as the accessorial structure (S. Ke et al. 2014). In searching geometry, (Witayangkurn et al. 2012) looked up in SR-tree to find the nearest polygons which works on Sphere/Rectangle based searching function. Proper indexing function is required for faster and efficient implementation of map matching algorithm.

Map matching is the process of aligning a sequence of observed user positions with the road network on a digital map. (Lou et al. 2009) proposed ST-Matching considers the spatial geometric and topological structures of the road network and the temporal/speed constraints of the trajectories. Based on spatio-temporal analysis, ST-Matching constructs a candidate graph from

which the best matching path sequence is identified. ST-matching algorithm have found to be significantly outperforms incremental algorithm in terms of matching accuracy for low-sampling trajectories. (Pink and Hummel 2008) incorporated road network topology in the matching process using a hidden markov model with introducing constraints for vehicular motion in an extended Kalman filter and reconstructing the original road network from the digital map using cubic spline interpolation. (Newson and Krumm 2009) introduced algorithm based on the hidden markov model that explicitly accounts for measurement noise and the feasible routes through the road network. Map matching algorithm uses a hidden markov model to find the most likely road route represented by a time-stamped sequence of latitude/longitude pairs. The algorithm was tested on ground truth data collected from a GPS receiver in a vehicle and was found to be robust to location data that is both geometrically noisy and temporally sparse. However, in vector-based map matching, the algorithm utilized heuristic method for aligning sequence of observed GPS point on to the road link segments. The vector-based algorithm searches for nearby links to the GPS points and if the distance between the projected points on that link and the actual point is less than matching error the points were added to array of possible map segments. Then map matching point which was previously identified is used as starting point and next point as the ending point of the GPS vector (Hadachi and Kibal n.d.). In addition, local path searching algorithm also adopted heuristic information to reduce search space. The method is implemented by construction square confidence region centered position fix from GPS that determined candidate matching path and with vehicle heading and the distances between position fix to links were integrated to match floating car data to the correct road segment (Chen et al. 2011). An online map-matching algorithm based on the hidden markov model (HMM) is found to be effective and robust to noise and sparseness as it focused on two improvements over existing HMM-based algorithms that are the use of variable sliding window (VSW) method and the novel combination of spatial, temporal and topological information using machine learning (Goh et al. 2012). (Raymond et al. 2012) also adopted map matching algorithm which is based on the ideal hidden markov model which used emission probability, state transition probability with initial state probability to find the sequence of roads that corresponds to the given sequence of raw GPS points. (Ren and Karimi 2009) presented a novel map matching algorithm to estimate wheelchair location in sidewalk networks also based on hidden markov model. The HMM-based map-matching algorithm matches with high accuracy GPS data to segments based on finding an optimal compromise between GPS data and

topological structure. However, some GPS points still gets mismatched due to failure in differentiating between the two sides of narrow roads. (Szwed and Pekala 2014) implemented a new incremental map-matching algorithm, which determine the vehicle trajectory by constructing a sequence of hidden-markov models (HMMs). In this method HMM state were also corresponding to a road segment and the sensor reading to an observation in HMM. A look ahead map matching (LAMM) implemented arc look-ahead which is a common technique in incremental map matching algorithms, used the road network topology to select future candidate paths by branching out n number of arcs from the arc that was previously selected for a match (Weber et al. 2010). With this an interactive voting-based map matching algorithm (termed IVMM) in which a voting process among all the sampling points to reflect their interactive influence was implemented for map matching operation in which each sampling point, their candidate road segments were determined, and for each candidate, there exists an optimal path which is passes through it. Every candidate will vote for their “best path”, and the global most optimal path was be chosen according to the voting result (J. Yuan et al. 2010). In addition to the probabilistic approach trajectory searching and matching was implemented using the Hausdorff distance; which is commonly used for determining the similarity between two set of point. An incremental algorithm was used for searching similar trajectories based on the Hausdorff distance in which the collection of trajectories was represented by R-tree indexes (Nutanong et al. 2011).

Taxi service provides one of the important mean of mobility in the urban population, however, the services itself is marred by many issues related to both driver and passengers (Peungnumesai et al. 2017). Also, the quality of service provided to the customers by the taxis are subject to the assurance to the customer which is affected the frequency of the taxi usages as well. It is also recommended that to improve the confidence level and service level for the taxi services the relevant governing company should provide knowledge and skill to the taxi drivers (Techarattanased 2015). Service model developed could enumerate the variable that would characterized the taxi service (Salanova et al. 2014), which in turn could be utilized for the decision-making process for the better taxi service operation. In computer modeling, the term “model” describes the abstract or simplified representation of a real world that is already present or planned for future. The simulation model is typically defined as a mathematical process or an algorithm that depends on various input parameters, which when processed with mathematical

expressions will result in one or more than one output, encapsulating the behavior and performance of a system in real-world scenarios (Abar et al. 2017; Raychaudhuri 2008).

Taxi service simulation is a dynamic process involving changing demand and supply as well as urban traffic environment which suggests stochastic behavior of taxi services that governs the movement as well as the distribution of taxis (Deng and Ji 2011; Maciejewski et al. 2016). Taxi customer bilateral searching and meeting behavior in a network was proposed in (K. I. Wong et al. 2005), which considered stochastic micro-searching behavior of both taxis and customers when they are searching for each other based on customer origin-destination (OD). The model featured location variation in the level of taxi services and stochastic microscopic searching behavior such that the taxi searched for passenger locally in the network that incorporated Markov chain approach as a route for which transition probability or the link choice probability was specified by the customer pick up rate within the network. An hourly zone based origin-destination matrix with the occupied vehicle was developed for evaluating the taxi service behavior which was then implemented for evaluating time-based taxi demand and supply concerning given location (Bischoff et al. 2015).

A probabilistic based model for time-dependent taxi behavior on a road segment as well as parking space was devised for taxi passenger recommendation in which probability of picking up a passenger was estimated when the taxi went for a specific parking space (J. Yuan et al. 2011). The model was primarily a recommendation system used for suggesting the taxi driver with a location, towards which they would pick up a passenger. Moreover, (B. Li et al. 2011) proposed passenger finding strategies based on large real-world taxi data which utilized two passenger finding strategies which were looking or waiting for a passenger that was analyzed using average pickup number over the given period and location. The model focus was also predicting potential passenger for the event before pick-up and after drop-off only. A time-dependent taxi behaviors model was proposed which incorporated taxi picking up, dropping off, cruising and parking system for both taxi drivers and passengers. The model was also primarily a recommendation system that was developed considering the queue length at parking place along with including day type, weather condition. The model provided some top parking places along with routes to them, given the current location and time of the taxi driver or a passenger (N. J. Yuan et al. 2013).

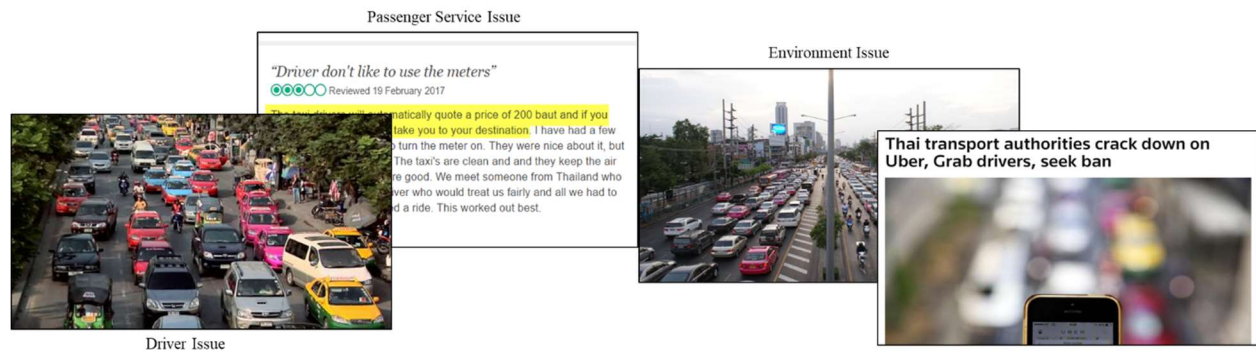
A time-dependent logit based search models were propped using global positioning data from an urban taxi in which profit per unit time was used as the factor characterizing taxi drivers search behavior (Szeto et al. 2013). A cell-based local customer search behavior was implemented for understanding vacant taxi behavior using a cell/grid-based approach which showed customer search decisions were significantly affected by the probability of successfully picking up a customer along the search route (R. C. P. Wong et al. 2014). The model was further improved by introducing discrete choice behavior representing taxi search behavior of taxi customers for hailing vacant taxis on the street was proposed by (R. C. P. Wong et al. 2015a) which adopted multinomial logit approach to model the preference of taxi customers of hailing vacant taxis on streets. Furthermore, study has been made for prediction model which employed learning algorithm to the GPS data. Real-time streaming data was implemented for predicting taxi passenger demand at a given taxi stand (Luis Moreira-Matias et al. 2013) in which the model predicted the passenger demand over taxi stand for a given period in future.

Existing taxi behavior model primarily focuses on finding passenger strategies through demand prediction with recommendation system. While only few studies are present that would provide insight of effect of an oversupply of taxi in the given area or vice versa (R. C. P. Wong et al. 2015b), the reason for which real taxi behavior modeling is important that would replicate the real-world system. The agent-based simulation model primary focus on understanding the real taxi behavior by utilizing GPS data from the probe taxi, from which further investigation could be made. Simulation model could be made for for efficient passenger finding system, managing taxi fleet operation i.e. optimizing the taxi service operation as well as understanding the impact of oversupply or undersupply taxi with respect to existing demand. Agent-based simulation in the field of computational science has proved to become a powerful tool for analyzing complex problem where random or stochastic behavior, as similar to the taxi behavior, can be presented together with behavioral rules. In this regards, (Grau and Romeu 2015) proposed a discrete event simulation model for modeling the behavior of agents operating in a city road network of which agents makes their own decisions for making trips. Similarly, (S. F. Cheng and Nguyen 2011) further provided a multi-agent based simulation which modeled taxi driver's strategies as a

decentralized discrete-event focusing on modeling only the taxi driver's behavior which was designed to make an aggregated pattern of taxi movement as similar to the real world.

### 1.3 Problem Statement

Though taxi service as considered as the fast and convenient way of commuting in the cities there are many issues related to its service. From the driver perspective it is working long hour with little income and taxi have to spend more on fuel and gas. As for the passenger perspective, taxi does not wants to go to the passenger destination place and taxi driver wants to charge fix rate. In addition there are lots of complain for the rejection from the taxi driver. These social issues the taxi service face in daily basis as depicted in Figure 1.1.



**Figure 1.1: Taxi operation issues in Bangkok**

The issues presented are basically the social issues that the taxi service faces in the Bangkok. The existing literature does shows that there are issues related with the taxi operation in Bangkok, Thailand whether it is from the driver perspective or it is from the passenger perspective. Spatial and temporal data are available from the taxi operation from the Bangkok and surrounding region. These data could be a value asset which could help improve the operation of taxi service through data mining technology. However there are technical issues also that are need to address the problem. The lack of available proper data infrastructure management system could be the hindrance if proper and efficient mining technology needs to be applied. In addition there are gap in the existing taxi behaviour model that this research focuses. The detail problem statement is presented as following.

**Proper Data Infrastructure Management System:** Without proper data infrastructure management system, the data mining working could be a very challenge task especially when dealing with the big data volume. Spatial data involving mobility data from the vehicle movement are constantly increasing. In such cases, how to properly handle the data becomes the primary task before any other data mining algorithm could be any applied.

**Taxi Operation Modeling with Quantitative Data Evidence:** The issue with the taxi operation services are exist as shown from the past literature. However, model to understand the behavior based on quantitative data evidence are not properly established yet. If the proper behavior model is not established it could pose a challenge when dealing with the ways to improve the service level of the taxi operations.

**Taxi Optimization Modeling:** The main two fundamental objectives of the taxi business or the service is to provide good service to the customer or passenger and in turn obtain the monetary profit. However, from the data evidence it is clear that there are issues related with both providing good service as well as getting better monetary profit. Taxi passenger are not happy when they are not provided with the good taxi service or when they are rejected to the service itself. On the other hand, taxi drivers are not getting enough passenger. Though the situation is ironic in nature itself, the problem does exist. One of way to minimize the issue is to optimize the operation of the taxi service. The optimization method as proposed will have ability for the driver in which driver can choose passenger depending upon the passenger origin and destination as well as available demand in the region. Optimization model is to provide recommendation to taxi driver which passenger would be better to choose and which not through mobile application. The hypothesis behind the model is that when driver have ability to choose the passenger then passenger rejection would be drastically minimized as well as choosing passenger would give driver some degree of freedom on how monetary profit could be improved. In addition, optimizing route for efficient taxi operation also plays an important role determining how much profit the driver can make by reducing the operation cost on fuel as well as its maintenance.



## 1.4 Research Objective

The main objective of this research is to help improve taxi operation in Bangkok region through quantitative data analysis from the GPS probe data from taxi. The overall objective is subclass as following

- Develop the data infrastructure management system for big mobility data handling and operation.
- Develop the taxi simulation model of the taxi operation for the Bangkok and the surrounding region
- Develop the optimization model for the improvement of the taxi operation

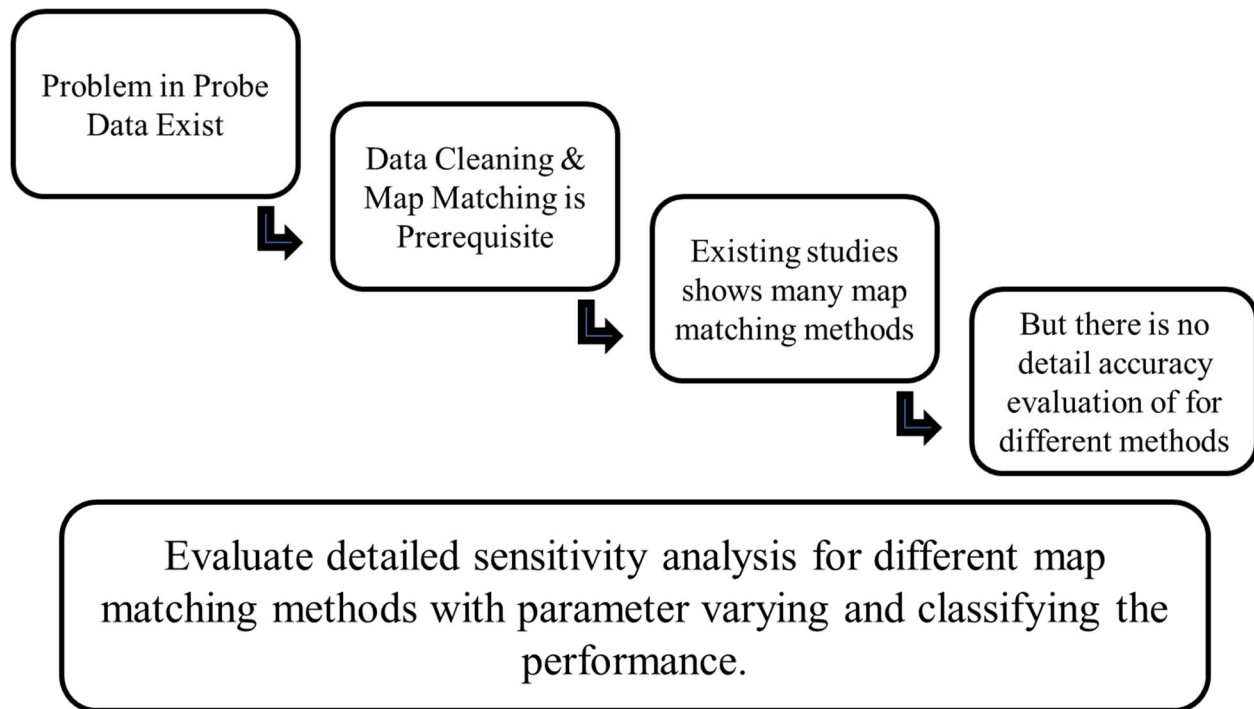
The objective is designed to address the issue for taxi operation in Bangkok which would provide the taxi driver with assistance that would increase the income, reduce the working hour in turn provide better service level for the passengers.

## 1.5 Key contribution

The key contribution of this research is in three categories that are framework for the data infrastructure, framework for the vehicle behavior simulation and framework for vehicle behavior optimization which are as described as follows and presented in Figure 1.2 and Figure 1.3:

1. Framework for the data infrastructure
  - Infrastructure development of the big data handling is provided which provided cloud-based platform that include features such as centralized architecture, interoperable, and importantly open to the users.
  - Road network infrastructure development with open street map which can be replicated to any country or cities where OSM data is widely available.

- Proposed different map matching algorithm which is based on geometry, topology as well as probabilistic approach. Measure of sensitivity analysis provided for different map matching algorithm based on road type as well as GPS data itself.



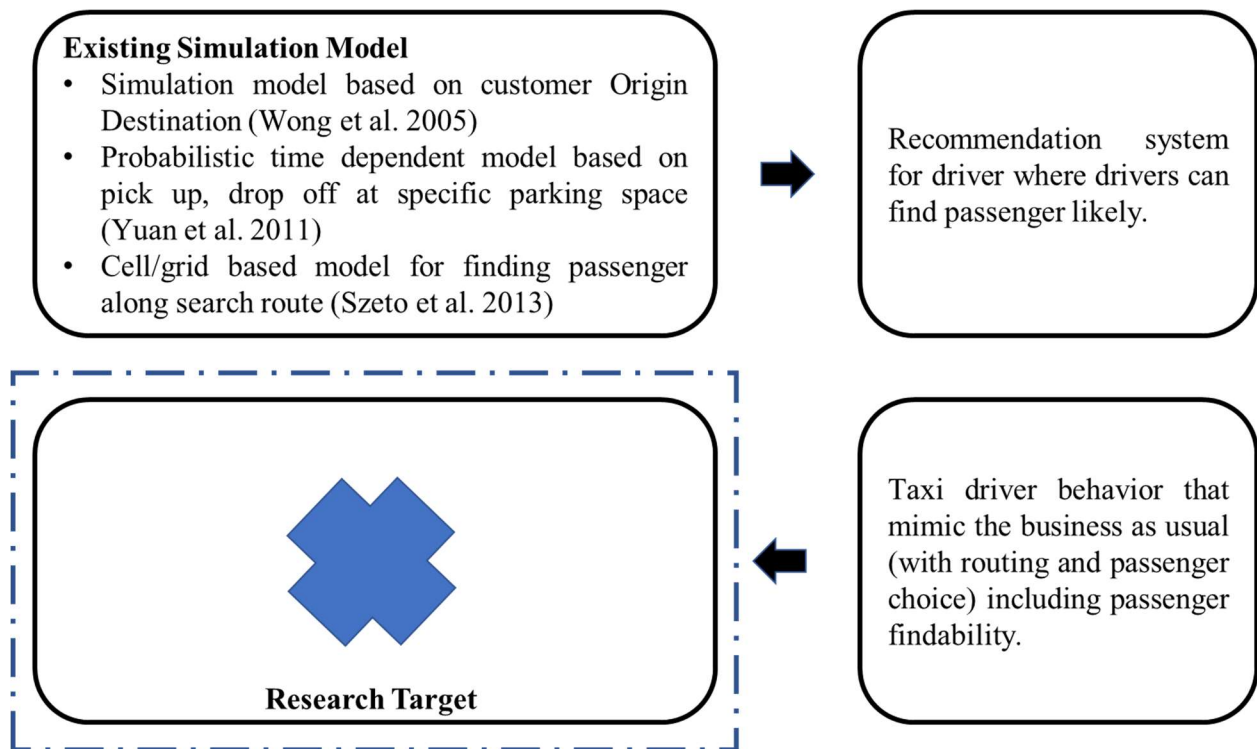
**Figure 1.2: Research contribution on data infrastructure**

## 2. Framework for vehicle behavior simulation

- Proposed a taxi agent regarding spatial and temporal domain based on a stay point cluster of probe GPS data and a kernel density of its timestamp.
- Formulated a concept of free taxi movement based on the movement direction of the taxi, which was introduced for searching passengers.
- Developed an agent-based simulation model which is based on multiple variable which includes taxi stay point cluster; trip information i.e. origin and destination; taxi demand

information; free taxi movement and network travel time which were derived from probe GPS taxi data.

- Utilized multiple spatial network data which are OSM road network and Grid network. Such that the agent's parameters were mapped into a grid network and the road network, for which the grid network was used as a base for query/search/retrieval of taxi agent's parameters, while the actual movement of taxi agents was on the road network, with routing and interpolation.



**Figure 1.3: Research contribution on simulation modeling**

### 3. Framework for vehicle behavior optimization

- Provide optimization of taxi operation based on driver's ability to make decision for choosing the passenger with optimized routing for better monetary benefit of the taxi drivers.

## 1.6 Research main tasks

The main research task is grouped into three main section that depicts the objective of the research work. The research task is illustrated in the Figure 1.2 as following. The three main research task includes ‘Data Infrastructure’, ‘Taxi Simulation Model’ and ‘Taxi Optimization Model’.

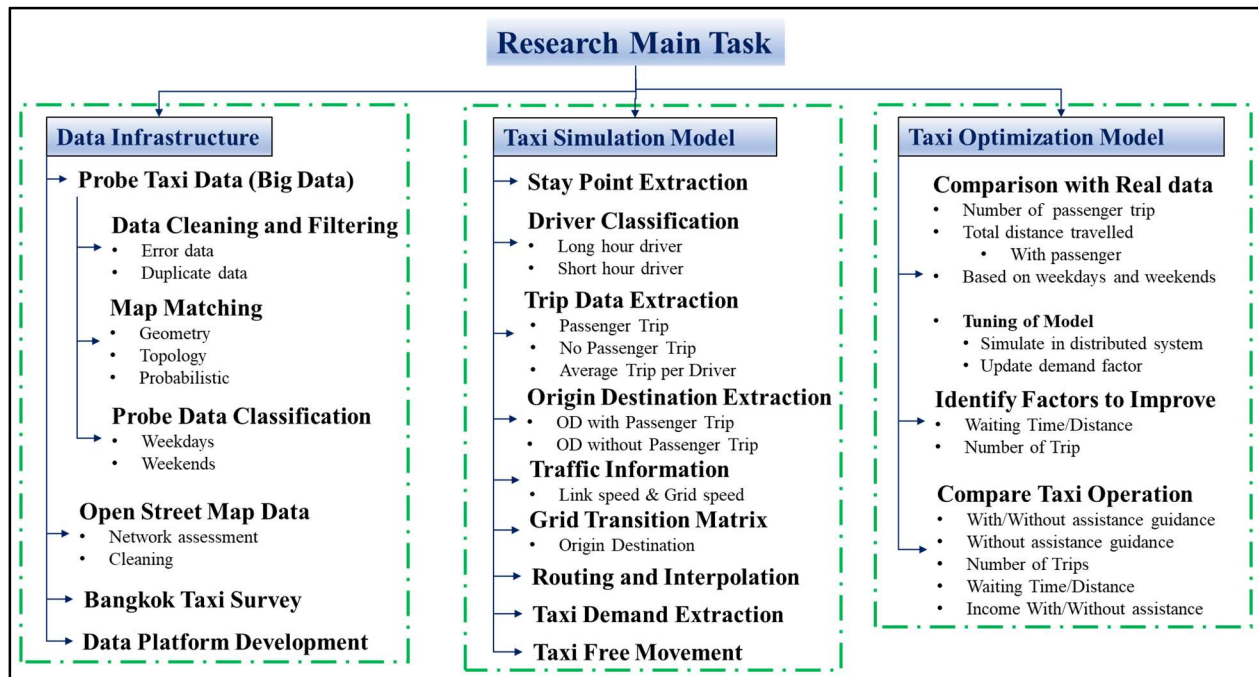


Figure 1.4: Research main tasks

The research task starts with the development of the data infrastructure. The data infrastructure is further categorized for handling probe data and road network data. Bangkok taxi survey is also conducted for the data preparation. Finally, a data platform namely Horton Data platform is developed to handle the big spatial dataset. The second research task includes taxi simulation model. The simulation model is constructed with multiple variables that are derived from the GPS probe taxi data. The third and the final task includes the optimization of the taxi operation based on proving assistance to the taxi driver to improve the overall taxi operation in the Bangkok and the surrounding regions.

## **1.7 Scope and limitation**

The scope of this research study stretches from data management, data manning to data mining for the social benefit. The data management is the fundamental entity in any big data operation. The proposed data management system is utilized for handling probe GPS data. However, the system is not limited to the probe GPS data only. The system is capable of handling and storing any data type and data structure. In this regard, the management system can provide support to other research activity as well. For the simulation modeling, the model is constructed through the data derived from the probe GPS data. Seemingly such model can be constructed without the need of the primary spatial data. The advantage of having such simulation model is that in the case the primary spatial data are too sensitive to be used for distribution to third party, the model could still be constructed with just the secondary spatial which would have deemed less sensitive in terms of privacy. As the matter or fact, the simulation model could be implemented in any cities not just from the probe GPS data but also from other spatial data from mobile phone or from Call Detail Record (CDR) data. As for the optimization model, model could also be implemented cities other than Bangkok as more and more countries are starting to collect probe data from the vehicles. In this regard, the overall system in this research could facilitate other research activity in terms of data management, data manning and data mining.

## **1.8 Structure of the thesis**

The main structure of the thesis is divided into three main part with 5 sub chapters as described below.

Chapter 1 provides the general background of the research. The focus of the chapter is to provide explanation on the existing issues prevalent and how the issue can be address as the objective for the research work.

Chapter 2 provides the detail development of various data infrastructure for handling big data, including network data as well as provide different algorithm for data cleaning.

Chapter 3 provides the detail development of the data driver taxi behavior modeling as well as way of validating the model with the existing dataset. Agent-based simulation and modeling is proposed for the modeling taxi driver behavior in which taxi behaves as an agent.

Chapter 4 provides the detail implementation of the optimizing the taxi behavior for improving the overall income of the taxi driver.

Chapter 5 provides discussion and conclusion for the overall research work.

## CHAPTER 2

### DATA INFRASTRUCTURE

2.

#### 2.1 Data Infrastructure Platform

##### 2.1.1 Data Infrastructure Development

When developing the data infrastructure there is always a question i.e. why we need a data infrastructure. The answer is quite simple, which is to handle the big data efficient and properly that would assist the research work. As shown here in Figure 2.1, raw data itself are huge in size and when secondary data are derived from it the size increases exponentially. Also, when working with spatial data a proper good quality map is required. These map data need to be processed through network assessment for cleaning and validating the topology. In adding the raw data, itself could be erroneous with noises, these data need to be cleaned properly. The data infrastructure could assist conduct task more efficiently as compare to working in a traditional system.

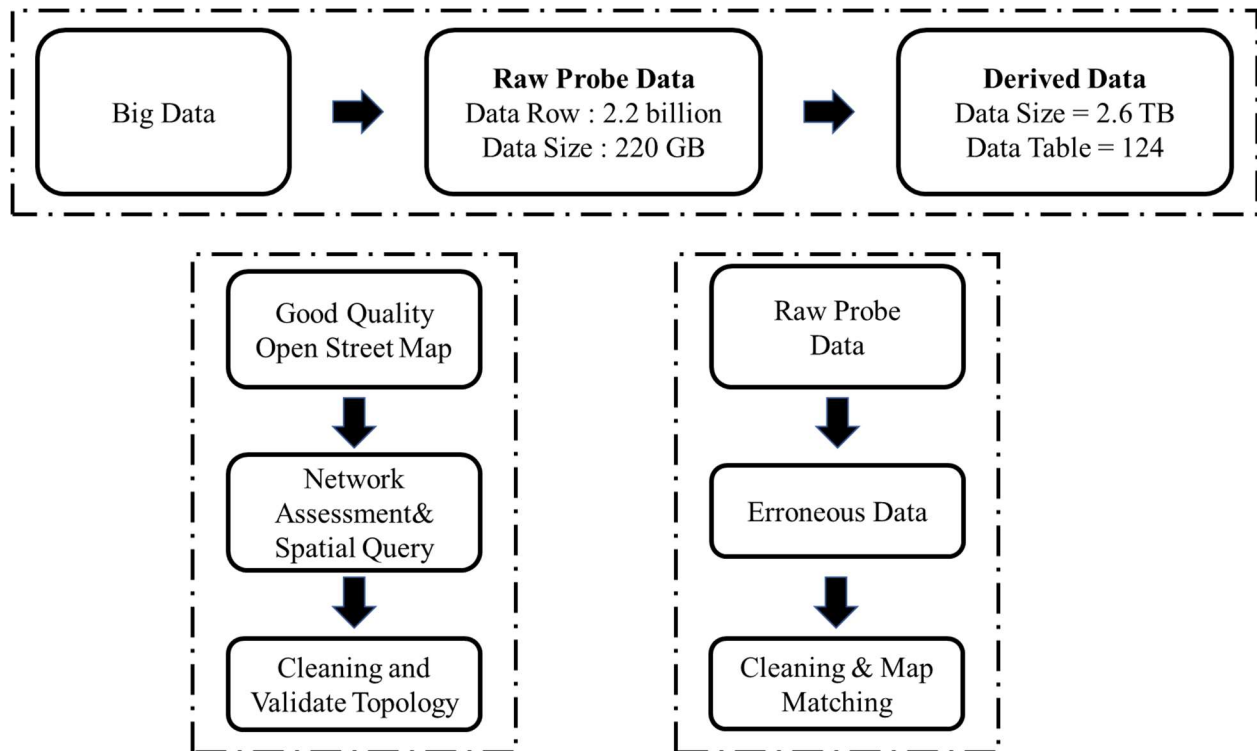


Figure 2.1: Need for data infrastructure

### **2.1.2 Big Data**

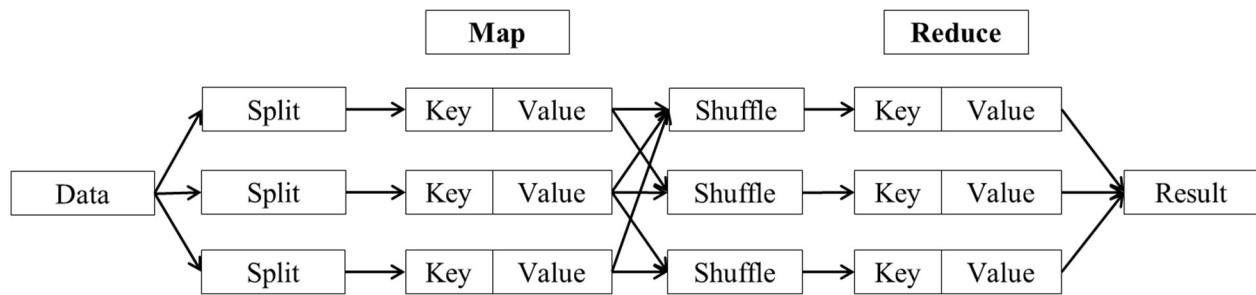
Big data recently have been the paradigm shift in the field of data science. The term Big Data are being used ubiquitously virtually any field from science and technology to finance market. So how do Big Data can be defined. A 3 dimensional data management terminology was introduced by (Laney 2001) regarding increase in data volume, data velocity and data variety which later became 3 V's in big data paradigm. Volume described the amount of data being generated. Velocity described how fast the data are being generated. Variety described different data type and data source including structured or non-structured data. (Schroeck et al. 2012) introduced one more V i.e. veracity that described the uncertainty and reliability referring to the quality of the data itself. (De Mauro et al. 2014) Quoted a proper definition is required that would make a path for systematic evolution of big data and defined big data regarding four key factors which were Information, Technologies, Methods and Impact. Regardless of the definition, big data itself possess many challenges. The challenges big data currently faces as described by (Intel 2012) in their report are data volume & growth, data infrastructure, data governance & policy, data integration, data velocity, data variety, data compliance & regulation and data visualization. One of important challenge that comes with the Big Data is the Big Data Infrastructure. Without proper data infrastructure data remains as only data for which a proper data infrastructure is required.

### **2.1.3 Hadoop Distributed System**

One of the prominent technology associated with the Big Data Infrastructure is the Hadoop System. Hadoop is a distributed computing platform designed to handle big data through parallel processing capability. The hadoop primary framework is MapReduce frame work with Hadoop Distributed File System as its file system (HDFS). The framework was which was originally published by Google in (Ghemawat et al. 2003). HDFS has the master slave architecture, where the master also known as NameNode manages the file system while the slave also known as DataNodes manages storage that run on it ("HDFS Architecture Guide" 2013). One of the reason hadoop has become powerful platform in the field of data science is that it has the capability of the horizontal scaling. Means that the processing power of the entire system can be increased just by



adding or increasing the DataNodes. MapReduce is the software framework for hadoop system (“MapReduce Tutorial” 2013) for parallel processing. The basis MapReduce (MR) architecture framework is shown in Figure 2.2 which operates on a <key, value> pair. The Map process divides the job submitted to the hadoop system into several identical tasks depending upon the number of available hardware for computation also known as data nodes. Each task is done in parallel processing. The output of the Map process after framework sort out process is fed to Reduce process. The Reduce process merges the data into single result.



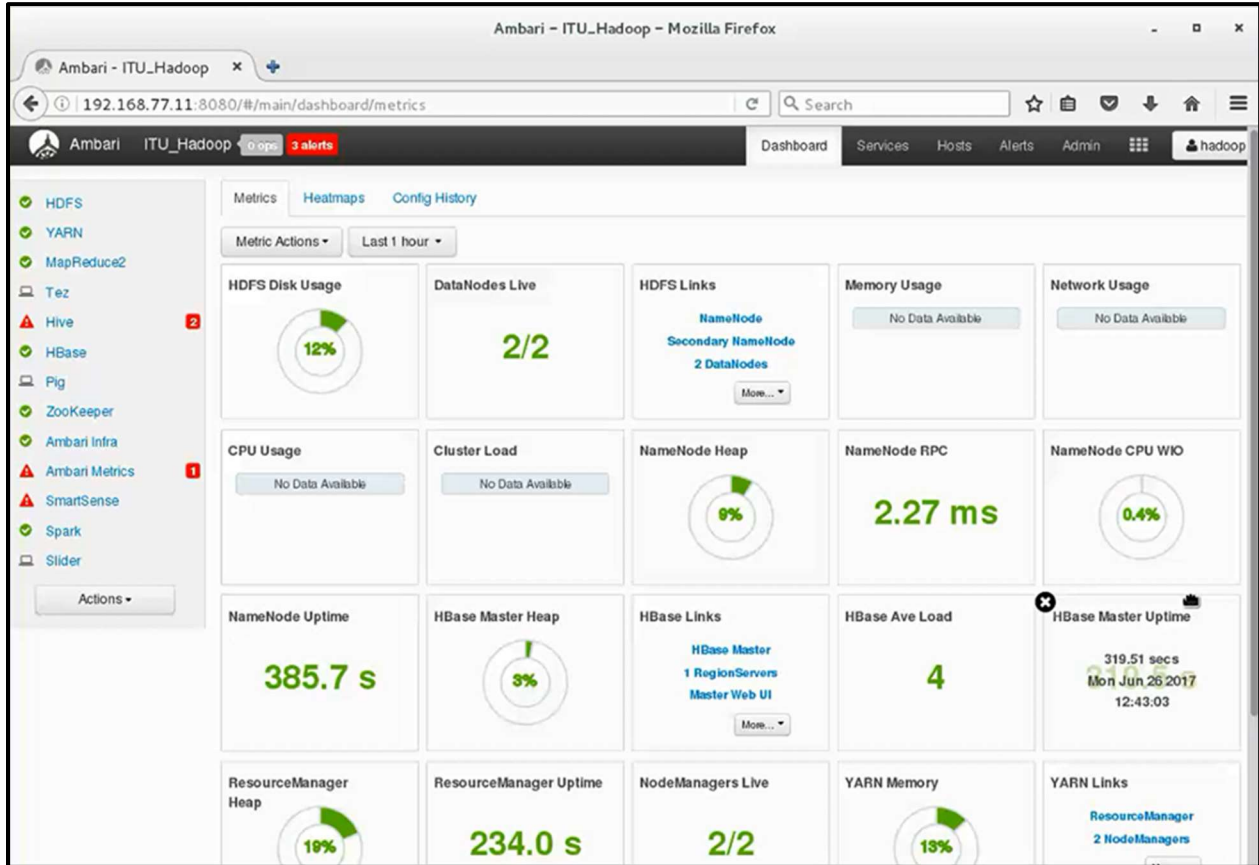
**Figure 2.2: MapReduce architecture of hadoop system**

In addition to the hadoop MapReduce framework, Apache Hive provides the data warehouse facility. Apache Hive is a data warehouse that provides capability of reading, writing as well as managing large dataset which is built on top of hadoop within the distributed file system (“Apache Hive” 2013). Hive provides the SQL like query known as HiveQL or HQL that which can be implemented in Hive CLI. The advantage of Hive over the MapReduce is that instead of writing MapReduce job for every task job could be implemented with the query. The utilization of Hive makes implementation of the job easier as Hive translates query to MapReduce job while running the job. Moreover, hive also features User Defined Function (UDF) and User Defined Aggregated Function (UDAF) implementation. The advantage of UDF and UDAF is that, in the case when there is no default function within hive, users can make their function as requirement within it.

#### **2.1.4 Horton Data Platform**

Hadoop and Hive systems provides scalability, flexibility when dealing with the big data. However, recently many data management project under Apache license are in development which provides

even greater advantage for handling big data. Some of the project are Apache Spark, Apache Tez, Apache HBASE, Apache Pig, Apache Zookeeper, Apache Storm. However, integrating and implementing different data management tool possess a challenge. Horton Data Platform (HDP) however allows users to work on different data management platform within a single distribution.



**Figure 2.3: Horton data platform interface**

Horton Data Platform is the enterprise-ready open source cloud based Apache Hadoop distribution which is based on centralized architecture YARN (“Horton Data Platform” n.d.). YARN is the architectural center for the hadoop enterprise that allows multiple data processing engine such as batch processing, SQL query, real time streaming to work in the single platform (“Apache Hadoop YARN” n.d.). Figure 2.3 shows the interface of the Horton Data Platform. The detail implementation of HDP is described in Appendix D. An example of the Hadoop/Hive query with user defined function and user defined aggregated function is shown in Appendix E.

## 2.2 Probe Vehicle Data

Vehicle probe data are data on vehicle behavior collected through a communication network (Nagashima et al. n.d.) which are being widely used for various Intelligence Transportation System (Liu et al. 2008). Taxi service is characterized as a fast and convenient way of commuting in big cities. As taxi are operational throughout the city, mobility data from these vehicles can be an asset for governing various urban management. Acknowledging the fact, Toyota Tsusho Nexty Electronics (Thailand) Co., Ltd, Bangkok, Thailand has equipped approximately 10,000 GPS devices onto the taxi running in Bangkok city and surrounding provinces. The probe dataset collected from the 10,000 GPS embedded taxi is at the sampling rate of 2 to 5 seconds. Figure 2.4 shows the basic working principle of the probe vehicle. The high sampling rate of the collection sizes, the average total data point collection could easily exceed more than 40 million every day. With the data being collected every day the data are increasing ever before. Data used for the research purpose is from 1 June 2015 to 31 July 2015 with the total data points of 2.2 billion data whose specification is shown in Table 2.1.

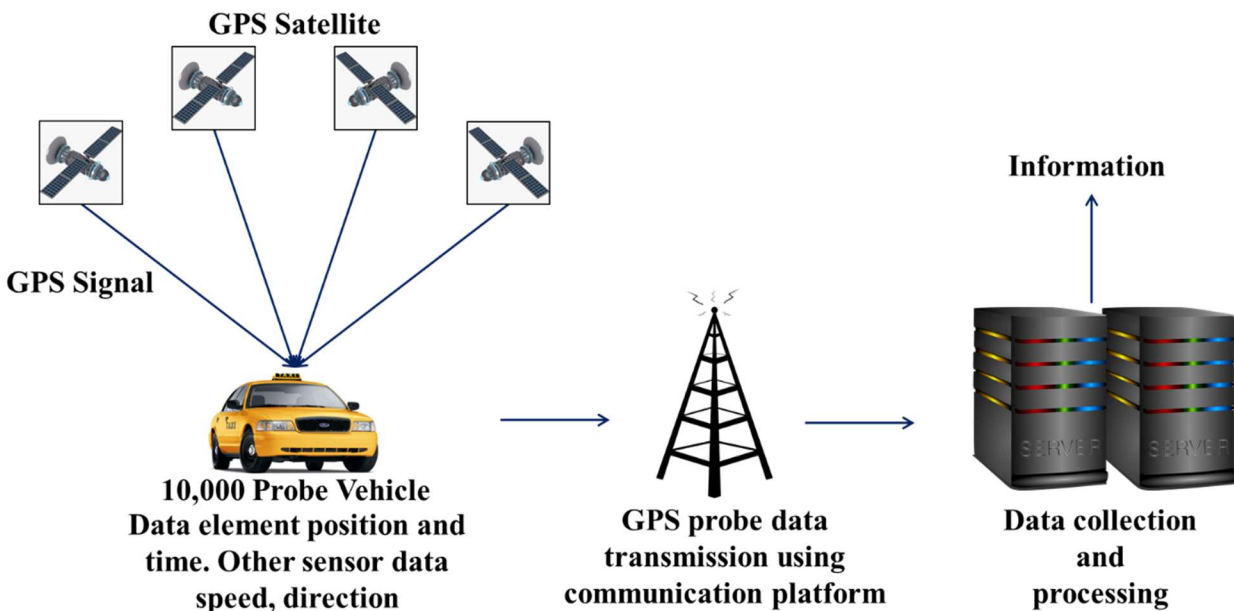
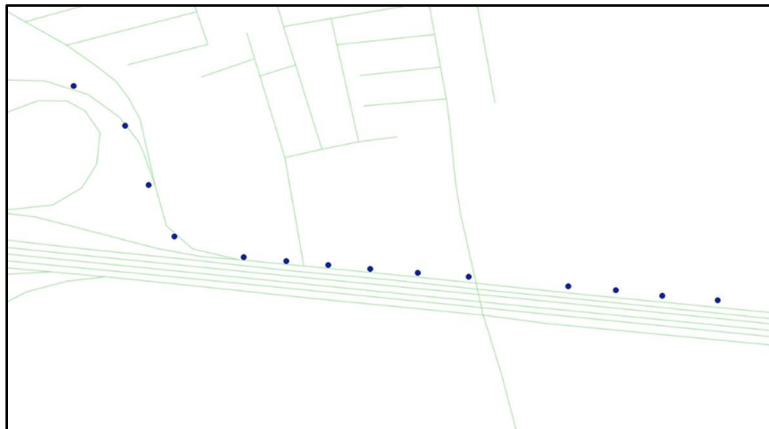


Figure 2.4: Probe vehicle working

**Table 2.1: Probe data specification**

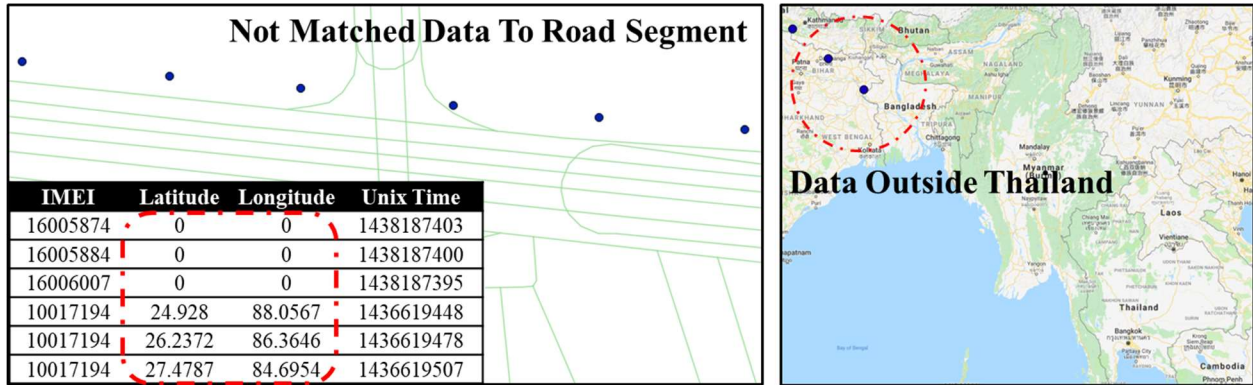
<b>Data Parameter</b>	<b>Description</b>	<b>Sample</b>
IMEI	International Mobile Equipment Identification Number.	10015646
Latitude	Geographic coordinate of the taxi regarding decimal degree.	13.749
Longitude		100.553
Speed	The speed of a moving taxi in km/hr.	42
Direction	The direction of a moving taxi in degree.	208
Error	Error status of for each GPS data point.	0
Engine	Engine status (0/1): 0 indicates the engine is off; 1 indicates the engine is on.	1
Meter	Passenger occupancy status (0/1): 0 indicate taxi with no passenger; 1 indicates taxi with a passenger.	0
Timestamp	Unix epoch timestamp. Time system which is described as a number of seconds elapsed since 00:00:00 coordinated universal time, 1 January 1970.	1388509240
Data source	Indicates the type of vehicle from which the data are being transmitted.	9



**Figure 2.5: Probe data example**

Figure 2.5 shows the taxi probe data example. However, as mentioned the raw probe vehicle data has error that need to be cleaned before future processing could be conducted. Figure 2.6 shows

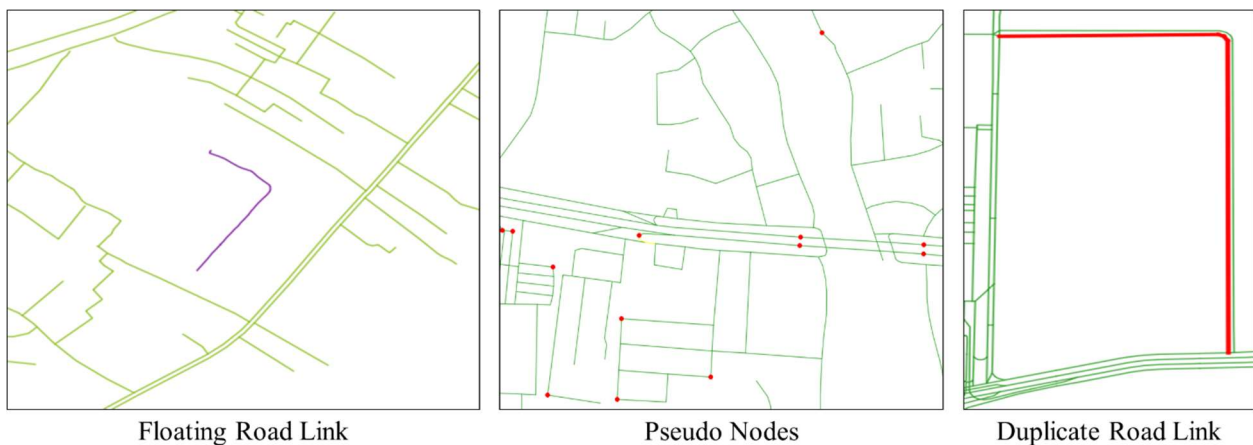
the type of different error the probe data contained. The raw probe data is from the Bangkok, Thailand. However, some of the data are outside of the Thailand. Some data the value is recorded as zero for both latitude and longitude. These are the completely incorrect data set that needs to be remove all together. In addition, the valid probe data are also not exactly not on the road network segment and hence need to be corrected.



**Figure 2.6: Probe data error example**

### 2.3 Road Network Data

The open street map data was obtained from (“geofabrik.de” n.d.) which has Protocolbuffer Binary Format (PBF) file type. The road network data was then generated by ‘osm2po’: which is a tool to convert OSM data into road network.

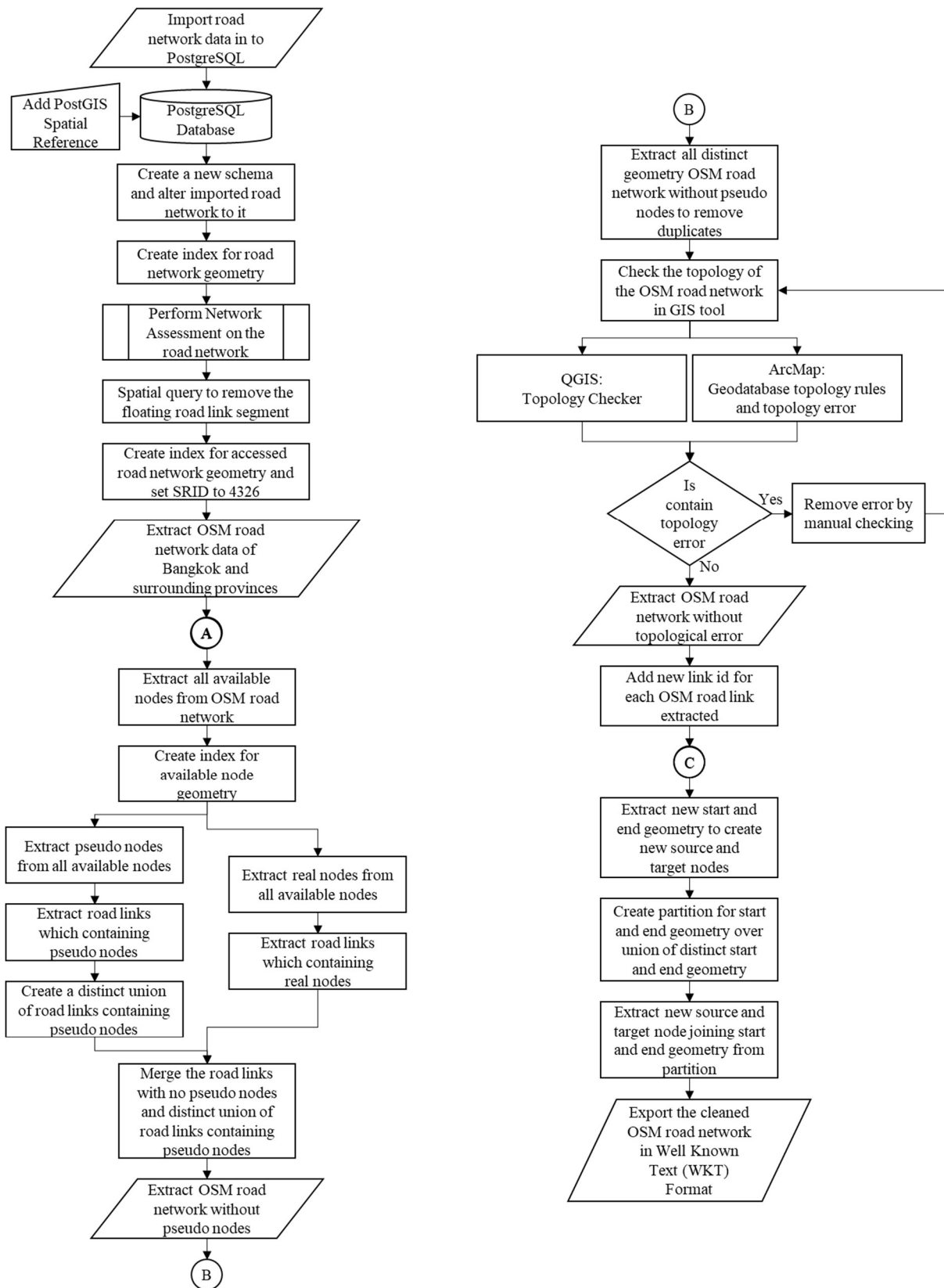


**Figure 2.7: Open street map topological error**

The extracted road network data, however contained various topological errors. The ‘floating road links: links that are not connected to the any of the major road network segments’. The ‘pseudo nodes: one or more than one false nodes that are present between the real road link nodes’. The ‘duplicate road links: same road link geometry appearing more than one time in the road network’. Figure 2.7 shows the various topological error links that were present in the road network.

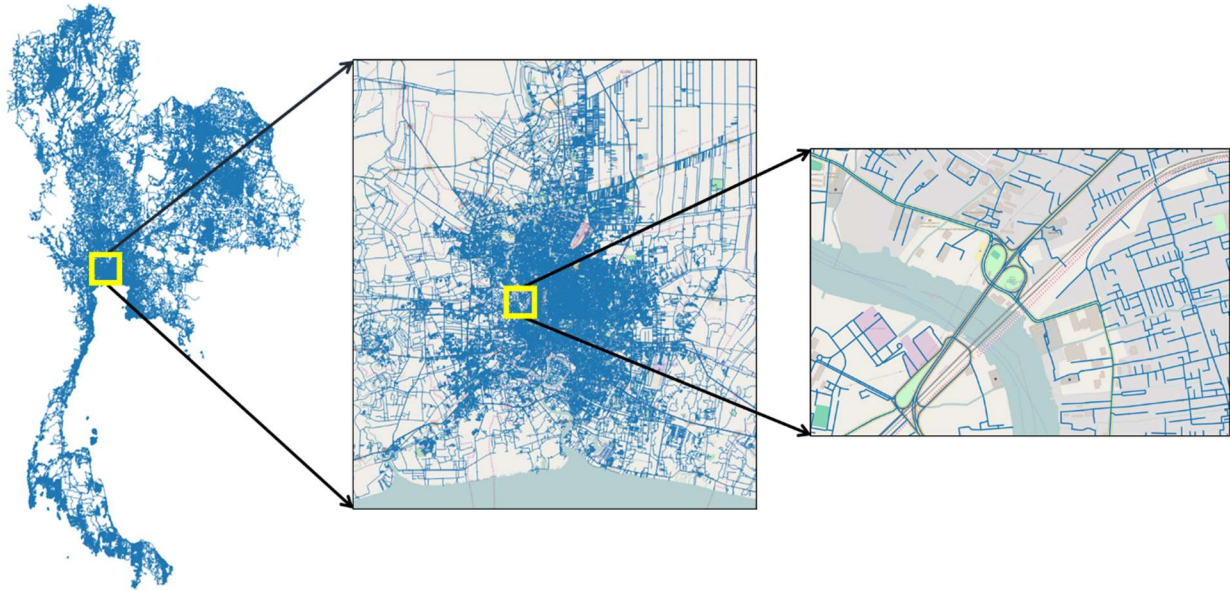
Before utilizing the open street map road network for map matching these topological errors were first cleaned. Cleaning the OSM road network is conducted with various queries in PostgreSQL in which PostGIS extension is used. Detailed flow diagram for cleaning topological error is shown in Figure 2.8. Road network data obtained from ‘osm2po’ conversion was first imported to the PostgreSQL database. Total of 1,107,798 road link feature were extracted for the whole of Thailand. For each road link, total of 19 parameters were generate that are ‘gid’, ‘id’, ‘osm\_id’, ‘osm\_name’, ‘osm\_meta’, ‘osm\_source’, ‘osm\_target’, ‘clazz’, ‘flags’, ‘source’, ‘target’, ‘km’, ‘kmh’, ‘cost’, ‘reverse\_co’, ‘x1’, ‘y1’, ‘x2’, ‘y2’.

A new schema was first created with for which the road network data was altered to it. Network assessment along with the various spatial queries were applied to the OSM road network to remove the floating road link segments as the first step of cleaning the topology. The road network was then cleaned for pseudo node error by first extracting and separating the road link with pseudo nodes and those without pseudo nodes. Union operation was then applied for those road link which contained the pseudo node. The process was iterated multiple times over those road link segments which contained more than one pseudo nodes. With pseudo nodes removed from road links, it was merged with the links that was originally free from pseudo nodes. As for cleaning the duplicate or overlapping road link, distinct geometry features in combination with feature within function was only selected. The obtained road network was then verified for topological error with GIS tool such as ‘QGIS: Topological checker’ and ‘ArcMap: Geodatabase topology rules and topology error’. Few topological error remaining were manually removed and verified until there was no topology error remained in the road network. With this a new id were assigned for each of the road link segment along with new source and target for each node of it. Finally, clean road network was exported in the Well-Known Text (WKT) format which to be used for map matching process.



**Figure 2.8: Detailed flow diagram of topological error correction**

Figure 2.9 shows the OSM road network extracted for the entire Thailand. As mentioned a OSM network provides us with the dense road network information, however for the data analysis purpose, the OSM from the Bangkok region and the surround provinces were only used.



**Figure 2.9: Open street map (OSM) Thailand road network**

## **2.4 Preliminary Data Analysis**

Preliminary data analysis included filtering out of various outlier data set along with duplicate position data and duplicate data from the probe dataset. Data analysis was conducted by first validating each parameter of each data rows as shown in Table 2.1. Outliner data such as invalid GPS points, GPS points out of bounding region of Thailand etc. were removed. Following this data were further cleaned out for ‘duplicate position data which were continuous set of same GPS positions data at different timestamp’ and for ‘duplicate data which were continuous set of same IMEI and timestamp’. Both duplicate position data and duplicate data were filtered out by comparing hash value of a current data row with hash value a previous data row. If the hash value of current and previous data row matched it was considered as duplicate and thus rejected else the data was kept for further processing. The dataset then subjected to the map matching which mapped probe data to the open street map road network.



## 2.5 Map Matching

Vehicle tracking data is an essential “raw” material for a broad range of applications such as traffic management and control, routing, and navigation. An important issue with this data is its accuracy. The method of vehicular sampling movement using GPS is affected by two error sources and consequently produces inaccurate trajectory data (Brakatsoulas et al. 2005). One of the aim of data infrastructure management is to analyze various map matching techniques that can be utilized efficiently and accurately for big GPS dataset. To match an original GPS tracking data to a digital map or a digital road network is often referred to as Map Matching. The general purpose of a map matching algorithm is to identify the true road segment on which a user (or a vehicle) is/was travelling (J. Yuan et al. 2010). As for a trajectory, it is defined as the path of a vehicle on a road network in which map matching is to estimate trajectory path from noisy position data. In Map matching the road map represents the topology of the road network and, since every point is localized, as it also provides a geometric representation (Mattheis et al. 2014). Map-matching is an integral part of various Intelligent Transportation Systems (ITS) including fleet management, vehicle tracking, navigation services, traffic monitoring and congestion detection. Such systems experience growing popularity due to proliferation of smartphone devices capable of receiving positioning data and transferring them over cellular networks (Szwed and Pekala 2014). The road network for the map-matching is obtained from the Open Street Map (OSM) which is cleaned for various topological error. The map matching process is performed considering road geometry, topology as well as the probabilistic approach.

Accuracy of the matched GPS points an essential parameter that determines the robustness of different map matching algorithm. Accuracy is defined as the fraction of correctly matched trajectory points in the ground truth path. A correct match is registered when the trajectory point is mapped to any road segment contained in the ground truth path (Goh et al. 2012). The local path-search map matching algorithm as mentioned in (Chen et al. 2011) provided the matching accuracy of 85.2% with point to curve method giving the accuracy of 82.4%. Map matching algorithm based on Hidden Markov Model yield the optimal accuracy of 92.1% in which evaluation was done for rural routes and urban routes, for which rural route was better than urban routes by a margin of 5% (Goh et al. 2012). (Newson and Krumm 2009) Mentioned map matching

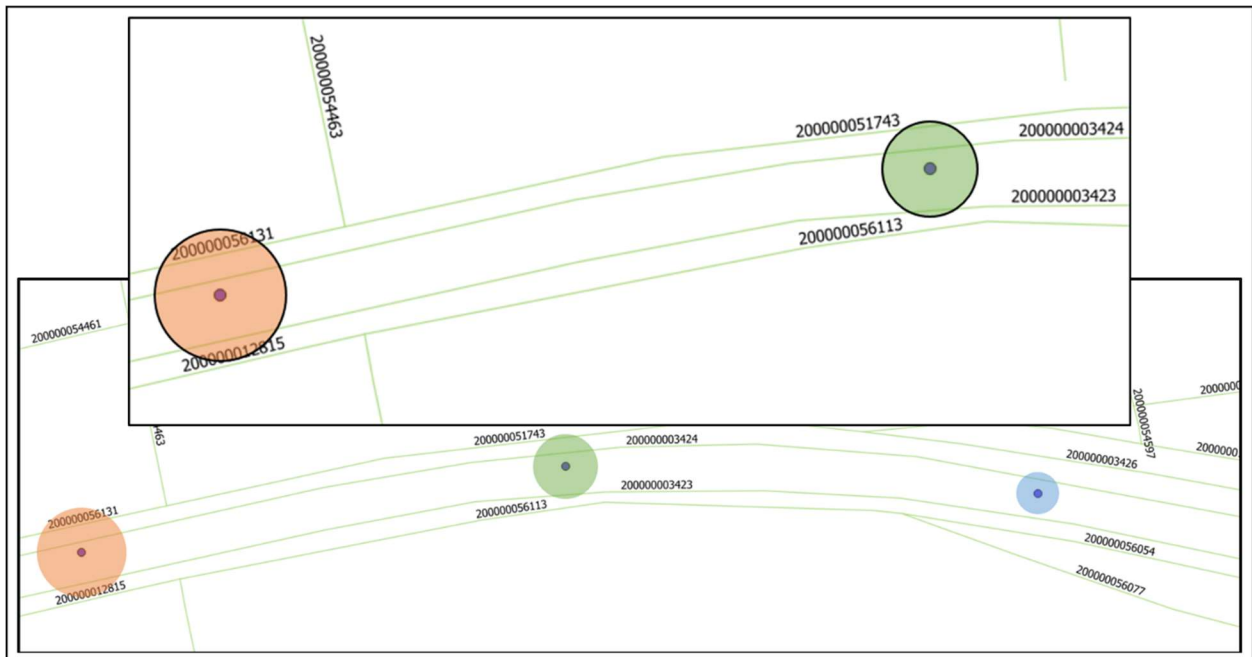
result ambiguous at the intersection with noisy measurement in which algorithm is based on Hidden Markov Model with overall error of 0.11%. In all these map matching algorithm emphasis, has been given more towards the performance of overall accuracy with little or no emphasis on the type of road network the algorithm has been much tested. A comprehensive accuracy assessment of map matching algorithm for various types of road network segment. In this regard, perform accuracy assessment for road segment such as ‘single lane road’, ‘multiple lane road’, ‘simple road intersection’, ‘complex road intersections’, ‘elevated road segment’ etc.

Moreover, the spatial and temporal data from the taxis are being collected from approximately 10,000 taxis with the sampling rate of 3 seconds or 5 seconds and with the dense open street map data, computational time for map matching becomes very high. To overcome this issue, a distributed computing platform for large scale data which utilized Hadoop/Hive that out performs other techniques and is also horizontally scalable was used. Hadoop can store large number of data and fast processing since it is a combination of multiple computer nodes. The main role of Hadoop is to archive large data including raw data and process the whole data (Witayangkurn et al. 2012) (Witayangkurn et al. 2013) (Mattheis et al. 2014). All computation involved in map matching operation is performed in a distributed system. Furthermore, the utilization of open street map as road link network, the techniques could be replicated and implemented for other country where OSM is available.

Map matching was performed considering geometry, road topology and probabilistic approach. Geometry map matching was done using buffer distance approach with the road link segment, Topology map matching was done by analyzing hausdorff distance similarity between GPS point sequence with road link segment and Probabilistic approach considered initial probability, measurement probability, transition probability of a consecutive GPS point. Before applying any map matching algorithm, it was first necessary to create an index of road network so that query and candidate selection within the road network becomes efficient and faster. STR tree which is Sort-Tile-Recursive algorithm was used to create index of the road network. STR tree index structure works basically for the two-dimensional spatial data which utilizes R-tree but has less overlap in between the nodes as compared to R-tree. Geometry of each road link as well as its parameter i.e. source, target and linkid was built in the index structure.

### 2.5.1 Buffer Operation

The simplest geometry operation for matching operation is a buffer operation which finds line segment from GPS point within a given buffer distance. In buffer operation, GPS point were searched in an index structure with pre-defined buffer distance. For each point the search result from the index produced multiple candidate road link. Distance between the GPS point geometry and candidate road link geometry were computed. Out of all the candidate, candidate with the least distance between itself and GPS point were considered as the matching road link. Finally, GPS point were projected on to the matched road link segment. In buffer operation, buffer distance was predefined and fixed in each operation however algorithm was tested for several buffer distances and the best optimal distance was identified based on accuracy assessment.



**Figure 2.10: Road link candidate selection at different buffer distance**

The buffer operation for map matching as shown in Figure 2.10 shows how variable buffer distance affects the candidate selection in this method. With smaller the buffer distance, fewer number of candidate were selected and vice versa. However, for each operation the buffer distance was predefined and kept constant starting from very small buffer distance to large buffer distance. In

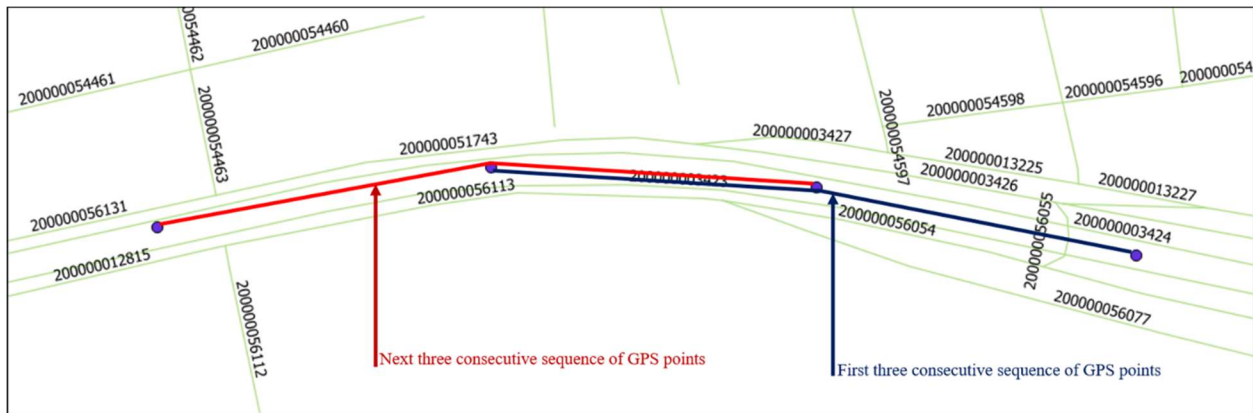
this method, ‘road link id’ and ‘projected point’ were returned as a map matched result. In the case when GPS points failed to identify any of the candidate road link, the original GPS point was kept as it is with ‘Not Available’ flag for the candidate road link.

### 2.5.2 Hausdorff Distance Similarity

Hausdorff distance matrix was implemented for topological operation which considered consecutive number of GPS points. A similar road link topology was searched that was defined by hausdorff distance similarity index to the GPS point sequence. By definition, hausdorff distance is a measure of the maximum of the minimum distance between two sets of objects (Nutanong et al. 2011). Given two finite point sets  $A = \{a_1, \dots, a_n\}$  and  $B = \{b_1, \dots, b_n\}$ , the Hausdorff distance is defined as in Equation (2.1).

$$H(A, B) = \text{Max } a \in A \{ \text{Min } b \in B \{ D(a, b) \} \} \dots\dots (2.1)$$

Set A is similar with Set B if every point in A is close to at least one point in B. The maximum of the distance from an element of A to its nearest neighbor in B. In this method similarity comparison was done between the set of GPS points with the road link segments.



**Figure 2.11: Sequence of GPS points for hausdorff distance analysis**

As per definition hausdorff distance similarity was computed between the set of finite point, thus the first set of point was built from several consecutive sequences GPS points. Number of GPS

point to be collected for sequencing was set at the beginning. A line segment was then constructed from the collected GPS points. Number of consecutive points were varied to find the optimum number of combination that gave the higher map matching accuracy. The second set of point were taken from the candidate road link segment as each road link segment are composed of finite set of coordinates connecting them. Road link candidate was then searched from the road network index which had similar topology as that of the point sequence. For each candidate road link, hausdorff similarity measure was computed and stored in the list. The road link candidate with the higher similarity measure was then identified, by sorting out the list from higher value to the lower value. The candidate road link with the highest similarity measure was considered as a matched road link. Process is continued for next consecutive sequences of GPS points by removing the first point and adding point to the end of previous sequence and so on.

The hausdorff distance operation as shown in Figure 2.11, shows a sequence of three consecutive GPS points considered. The first sequence took the first three GPS point then for the next sequence first point was removed and next point is added to end of previous sequence. The process was iterated until all the GPS point is considered. The returned result from this algorithm is ‘road link id’ and ‘projected point’ with the case in which road link candidate fails to identify, original GPS point is kept as it is. In the case when candidate road link failed to be searched, original GPS point was kept as it is with ‘Not Available’ flag for the candidate road link.

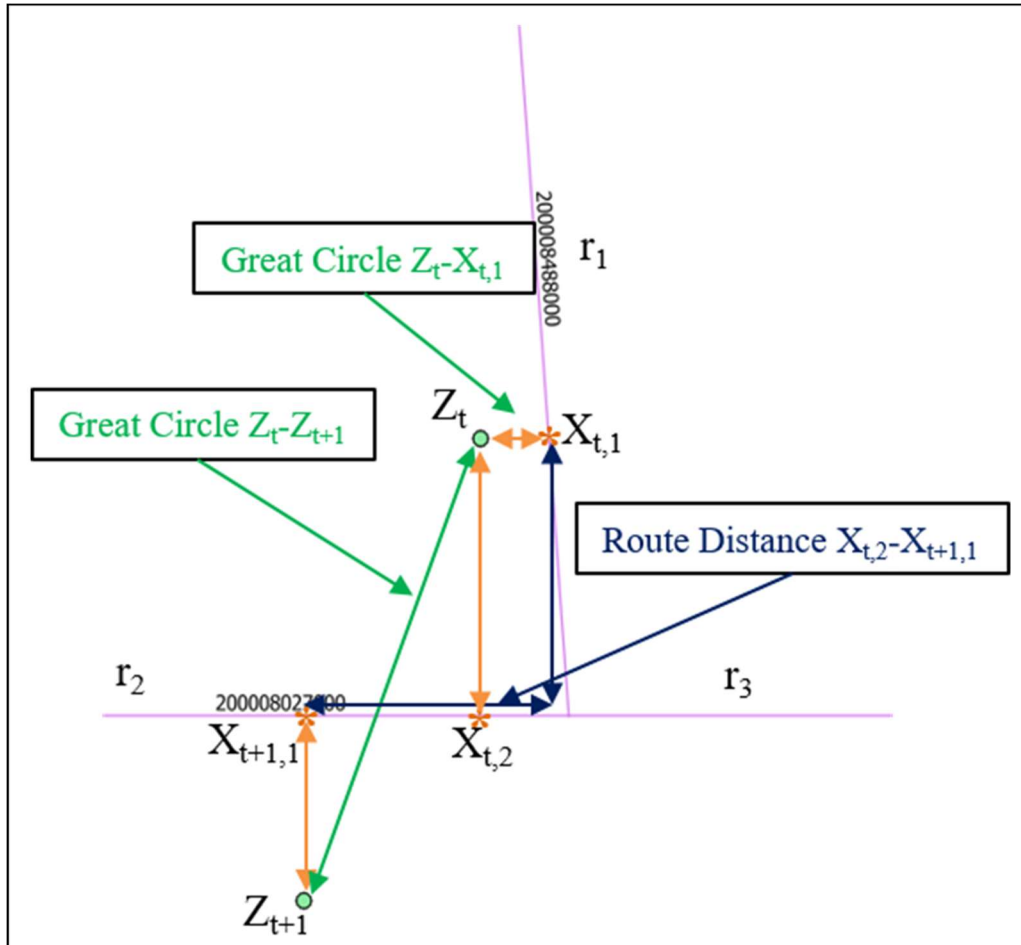
### **2.5.3 Probabilities Map Matching**

The probabilistic approach for map matching was implemented considering three probabilities i.e. initial probability, measurement probability and transition probability. Figure 2.12 shows the example for terms used for compute various probabilities.

#### **Term Definitions**

Road link segment: Candidate road link segment are represented by  $r_i$  and  $r_j$  where ‘i & j’ = {1,2,3...n}. GPS points: GPS points are represented by  $Z_t$  &  $Z_{t+1}$  where t represented each timestamp. Projected Coordinates: Projected coordinate are represented by  $X_{t,i}$  &  $X_{t+1,i}$ . Great

circle distance: Shortest distance between two points on the surface. Route distance: Distance between two consecutive projected coordinates through the road link segment.



**Figure 2.12: Probabilistic map matching term definition**

Initial probability: Initial probability was used for ranking each of the candidate based on the Euclidean distance between the GPS point  $Z_t$  and candidate road link segment  $r_i$  selected from searching the road network index. Initial probability is represented by Equation (2.2).

$$P(Z_t \rightarrow r_i) = \exp(-(|Z_t \rightarrow r_i|)_{euclidean}) \dots \dots (2.2)$$

Measurement probability: Measurement probability, also known as an emission probability shows probability of an observation which is  $Z_t$  such that the GPS points is on the road link segment  $r_i$ .

Term  $\sigma_z$  is the standard deviation of the GPS measurement (Newson and Krumm 2009), however  $\sigma_z$  was set empirically in this paper. Term  $|Z_t \rightarrow X_{t,i}|$  represented the great circle distance between the observed GPS point  $Z_t$  and projected coordinate point on the road link segment  $r_i$ . Measurement probability is represented by Equation (2.3).

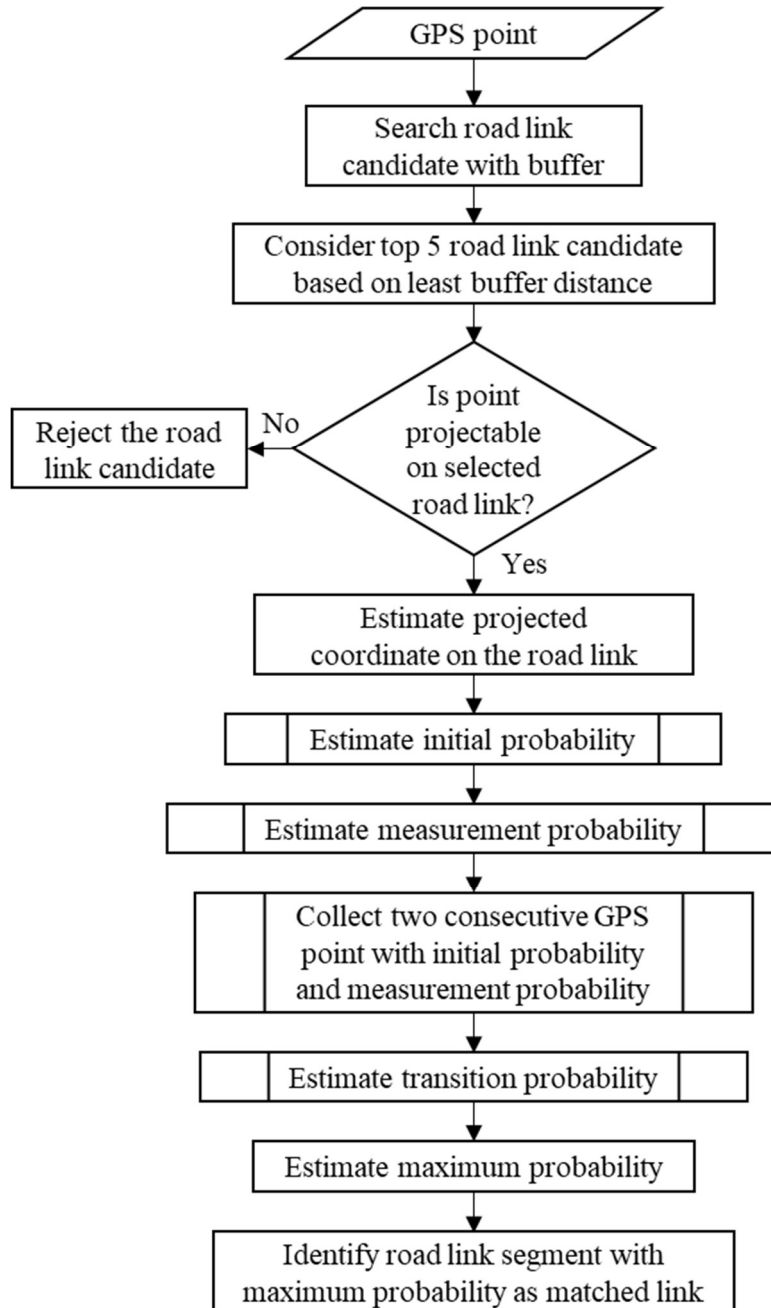
$$P(Z_t|r_i) = \frac{\exp\left(-0.5 \times \left(\frac{(|Z_t \rightarrow X_{t,i}|)_{Great\ Circle}}{\sigma_z}\right)^2\right)}{\sqrt{2\pi} \times \sigma_z} \dots\dots (2.3)$$

Transition probability: Transition probability showed how likely the GPS observation point moved between the candidate road link segment. The term  $|Z_t \rightarrow Z_{t+1}|$  represented great circle distance between two consecutive GPS point i.e.  $Z_t$  and  $Z_{t+1}$ . Similarly,  $|X_{t,i} \rightarrow X_{t+1,j}|$  represented the route distance between the projected coordinates of the observed GPS points  $Z_t$  and  $Z_{t+1}$  on to the road link segment  $r_i$  and  $r_j$  respectively. The parameter  $\beta$  is a probability parameter as describe in (Newson and Krumm 2009), however  $\beta$  was also set empirically. Transition probability is represented by Equation (2.4).

$$P(X_t \rightarrow X_{t+1}) = \frac{\exp\left(-\frac{(|Z_t \rightarrow Z_{t+1}|)_{Great\ Circle} - (|X_{t,i} \rightarrow X_{t+1,j}|)_{Route}}{\beta}\right)}{\beta} \dots\dots (2.4)$$

The map matching approach as shown in Figure 2.13 shows the flow diagram for conducting the map matching using the probabilistic approach. For each point at first candidate road link segment was searched from the index using buffer operation. To reduce the computational complexity only five candidate road link segment was chosen based on the least distance from the point to the candidate road link. Each candidate road link was tested whether a valid projection could be made or not. If valid projection could not be made those candidate road links are removed otherwise GPS points were projected on to it. With projected coordinates estimated, initial probability followed by measurement probability were calculated using the Equation (2.2) and (2.3). The process of estimating projected coordinate, initial probability and measurement probability was done for the next consecutive GPS point. With the information, projected coordinates two

consecutive GPS points, transition probability is estimated using Equation (2.4). Maximum probability from three probabilities is then computed that identifies the matched road link segment.



**Figure 2.13: Flow diagram of map matching with probabilistic approach**

Table 2.2 shows an example of a map matching operation using the probabilistic approach. In this example two points are selected i.e. Point A and B respectively. For both point A and B, there are



three candidate road link, however maximum of five candidate road link are possible depending upon the search result. For each candidate road link at first initial and measurement probabilities are computed represented by IP and MP in Table 2.2. Transition probability, represented by TP, between candidate road link of Point A and Point B i.e.  $TP\{C \Rightarrow NPC\}$  was then computed. From these three probabilities, maximum probability is computed as shown in Equation (2.5). The road link candidate with maximum probability is the matched road link.

$$P_{max} = Max\{P(Z_t \rightarrow r_i) * P(Z_t|r_i) * P(X_t \rightarrow X_{t+1} )\} \dots\dots (2.5)$$

**Table 2.2: Probabilistic map matching example**

Point A: 14.21763, 100.68918				Point B: 14.21763, 100.68918
Candidate(C)		NPC	TP	Candidate(C)
200000183119	IP = 0.9998 MP = 0.01261	200000183118	0.0344	200000183118
		200000183115	0.0316	
		200000183136	0.000699	
200000183144	IP = 0.9998 MP = 0.01222	200000183118	0.0265	200000183115
		200000183115	0.0244	
		200000183136	0.0085	
200000183142	IP = 0.9999 MP = 0.01367	200000183118	0.0326	200000183136
		200000183115	0.03045	
		200000183136	0.0343	
Point = Observed GPS Point				
C = Candidate Road Link				
NPC: Next Point Candidate Road Link				
IP = Initial Probability				
MP = Measurement Probability				
TP = Transition Probability				
Matched Link: 200000183142 For Point P: 14.21763,100.68918				





Normal (single lane) road link



Multiple lane road link



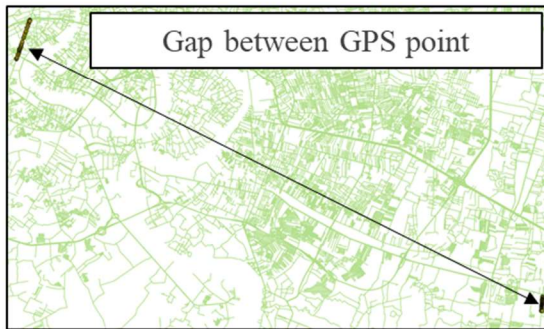
Simple road intersection



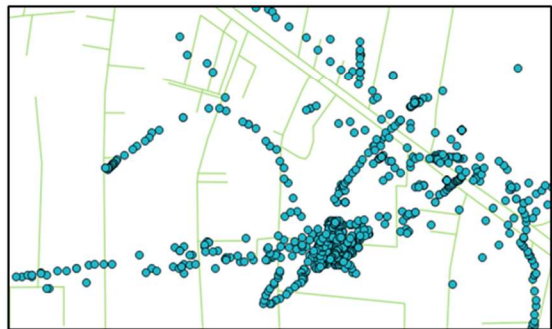
Complex road intersection



Elevated road link

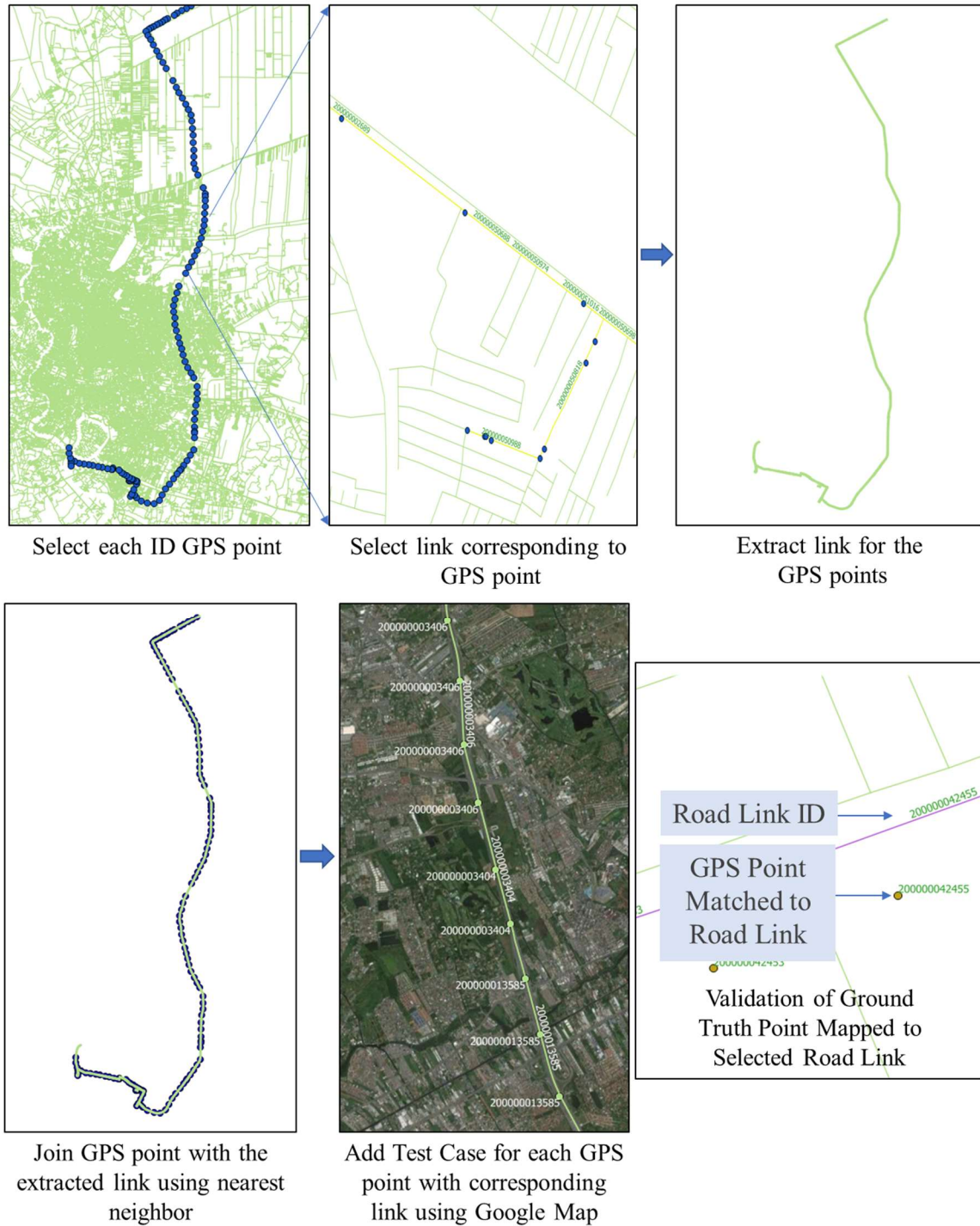


Large gap between GPS points



Random jump of GPS points

**Figure 2.15: Different test cases including road type and GPS data type**



**Figure 2.16: Ground truth data preparation with test case**

Each selected ground truth GPS point was categorized per test case constructed based on where those points were located. For an example, a GPS point can have case 2 and case 5 i.e. that point in is multiple road link as well as in the elevated road link. Categorizing of all the ground truth

GPS point was done manually with visual inspection. At first GPS data set was separated and sorted out based on its IMEI value and timestamp. For each IMEI, road link corresponding to the GPS point was selected manually from the whole road network. The process of selecting road link was done using ‘select tool’ in QGIS. A road link trajectory was then extracted for each of the IMEI. This trajectory represented the true path of the GPS points. With this, each GPS points were map matched on to the extracted trajectory using nearest neighbor as shown in Figure 2.16. The mapped GPS point was then considered as a ground truth data set. Finally, test case was added for each GPS points with corresponding to various base map like google map, open street map etc. Table 2.3 shows the sample ground truth data set with test cases.

**Table 2.3: Sample ground truth data**

<b>IMEI</b>	<b>Latitude</b>	<b>Longitude</b>	<b>Mapped Linkid</b>	<b>Case Number 5</b>
10000023	13.61122	100.6584	200000026510	*
10000023	13.60455	100.6543	200000026509	*
10000023	13.60058	100.6475	200000026509	*

**Table 2.4: Distribution of GPS points with different test cases**

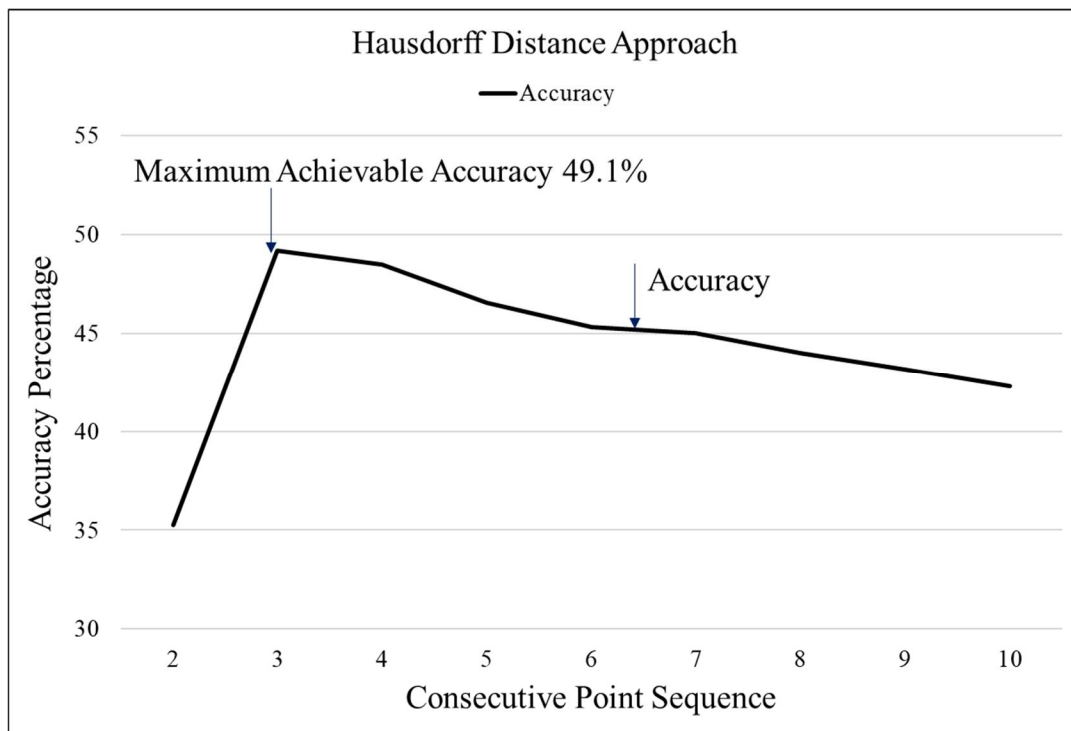
<b>Test Case</b>	<b>Number of GPS points</b>	<b>Percentage (%)</b>
1	2574	27.91
2	6639	71.98
3	283	3.07
4	875	9.49
5	1379	14.95
6	30	0.33
7	68	0.74
<b>Total number of GPS point in Case = 11,848</b>		
<b>Total number of ground truth GPS point = 9,224</b>		

The distribution of GPS points according its test case is shown in Table 2.4. Out of all Case 2 has highest number of GPS points with 71.98% followed by Case 1 and Case 5 with 27.91% and 14.95%. Having high distribution GPS point for case 2 shows that most of the roads have multiple

lane in region though the GPS points were randomly sampled. As for Case 6 and Case 7, it represented the least number of GPS point with 0.33% and 0.74% respectively. On note, total number of GPS points in Case is more as compare to total number of ground truth GPS point as single point can have multiple different cases.

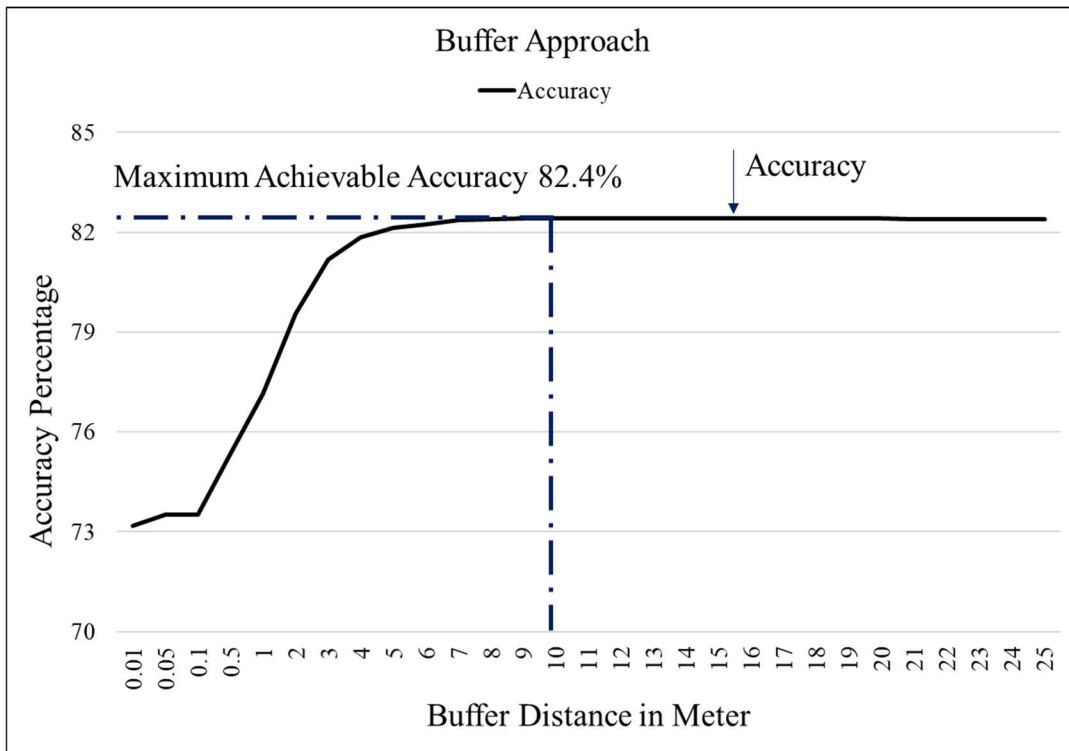
## 2.7 Map Matching Evaluation

The evaluation of map matching accuracy assessment showed probabilistic approach performed better as compared to hausdorff distance approach and buffer distance approach. Each of the algorithm was tested and compared by varying its parameter to get maximum achievable accuracy percentage. Out of all three hausdorff distance approach performed least with maximum achievable accuracy percentage of only 49.1%. With this approach half of the total ground truth points were either wrongly matched or not matched at all. The buffer distance approach, over all map matching accuracy increased with maximum achievable accuracy percentage of 82.4%. Finally, with probabilistic approach, maximum achievable accuracy percentage of 86.6% was obtained.



**Figure 2.17: Map matching accuracy with hausdroff distance approach**

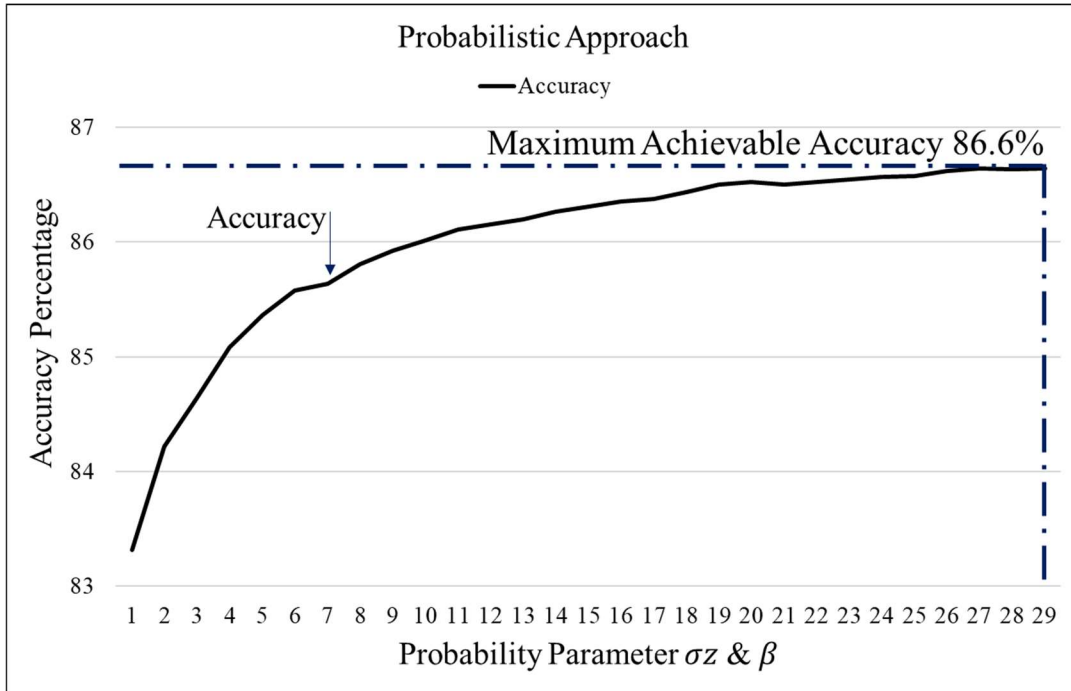
In the case of hausdorff distance approach, number of consecutive GPS point sequence was varied. The result showed, with two-point sequence the accuracy percentage was least and at three-point sequence accuracy was maximum. However, with increase number of point sequence after that, accuracy decreased gradually as shown in Figure 2.17. Since, road network was dense and not all GPS point were following the path that matches the shape of a road link, similarity was lower and thus the accuracy of matching.



**Figure 2.18: Map matching accuracy with buffer distance approach**

As for the buffer distance approach, parameter varied was the buffer distance. When buffer distance was less than a meter, accuracy percentage obtained was least. With buffer distance gradually increased from a one meter to ten meters, accuracy percentage also increased. However, with further increment beyond ten meters, there was no significant improvement in accuracy percent and remained constant as shown in Figure 2.18 suggesting that the buffer distance has reached the threshold value. This also suggested that most of the GPS points were in the ten-meter

buffer distance range. As for comparison buffer distance approach result was more significant regarding matching accuracy than that of hausdorff distance similarity approach.



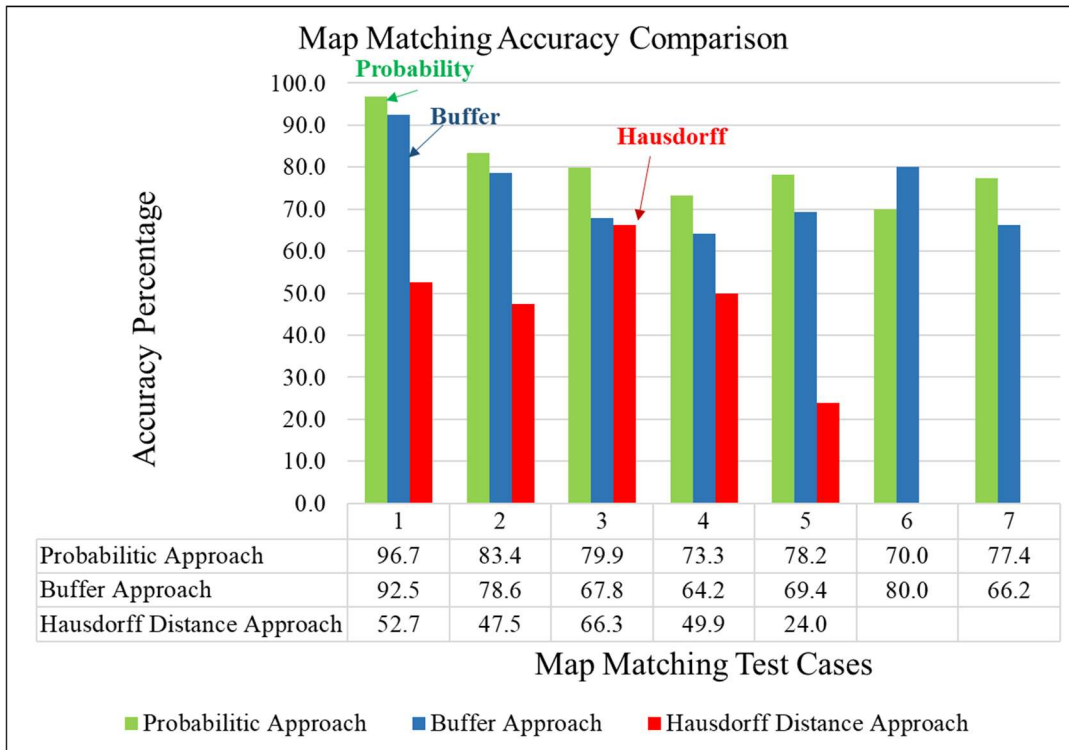
**Figure 2.19: Map matching accuracy with probabilistic approach**

Varied parameter for probabilistic approach was standard deviation  $\sigma_z$  and probability parameter  $\beta$ . With both parameter given same value and increment, accuracy percentage also gradually increased as shown in Figure 2.19. In this approach, threshold value was obtained when  $\sigma_z = 29$  and  $\beta = 29$ . Increment of parameter beyond threshold value did not increased overall accuracy percentage. The result from the probabilistic approach was better in terms of accuracy as compared to both buffer distance and hausdorff distance approach. In addition, probabilistic approach algorithm leaves room for further improvement in terms of transition probability computation. Here transition probability implemented was forward transition i.e. looking ahead of next GPS point. However, improvement could be made by introducing backward transition probability in addition to forward transition. Lastly, in all map matching algorithm, if GPS points failed to identify any of the candidate road link, the original GPS point was kept as it is with ‘Not Available’ flag for the candidate road link. The detailed test case of single probe taxi data is shown in Appendix A.



## 2.8 Map Matching Test Case Accuracy Evaluation

Accuracy assessment conducted for each of seven test cases also showed probabilistic approach performing better as compared to both hausdorff distance approach and buffer approach as shown in Figure 2.20.



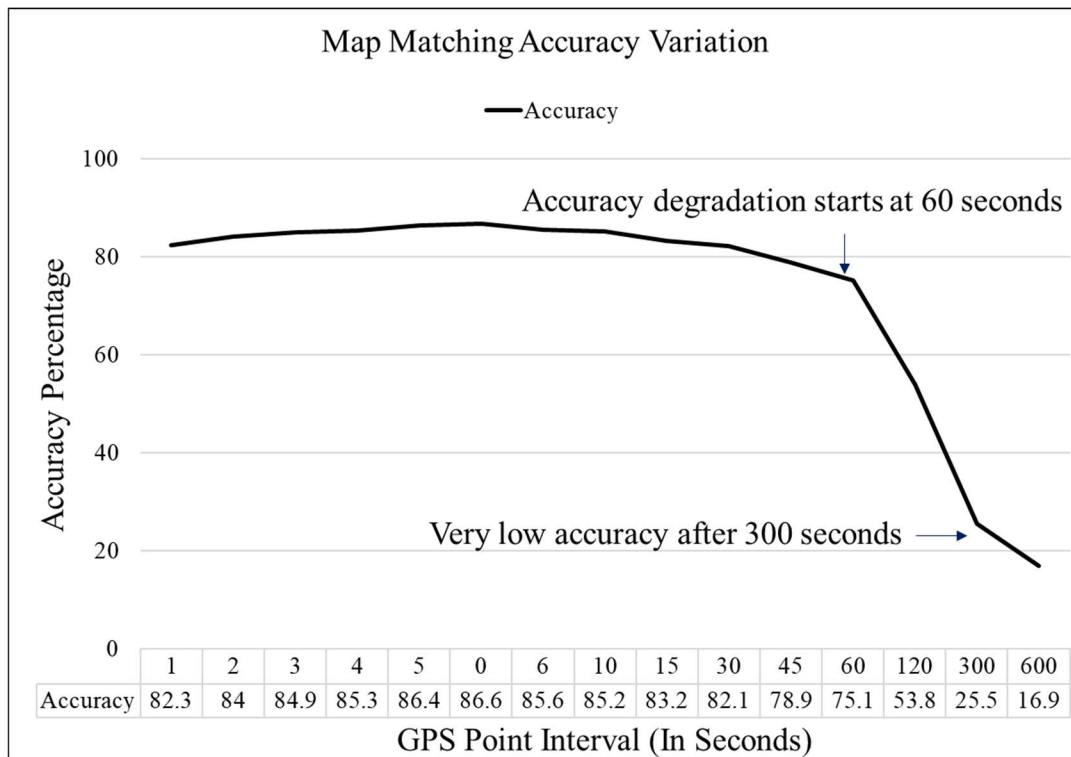
**Figure 2.20: Map matching accuracy evaluation for seven test cases**

For Case 1, probabilistic approach and buffer distance approach could get 96.7% and 92.5% respectively. However, hausdorff distance approach gave just about 52.7% accuracy only. As for Case 2, which contain about 71.98% of the GPS point, accuracy percentage was 83.4%, 78.6% and 47.5% respectively for three approaches. For Case 3-7, accuracy percentage was in 70% range for probabilistic approach. As for buffer distance approach accuracy was in 60% range except for Case 6 which had 80% accuracy and was also better than probabilistic approach. However, Case 6 has least number of GPS points associated with it, thus in overall performance, probabilistic approach outperformed the others. Moreover, hausdorff distance approach for Case 6-7 gave no

result at all in addition to lower accuracy percentage in other cases. The result clearly showed probability approach was a better algorithm for map matching. However, result also indicated that high matching accuracy are not always possible in every road link segments. In difficult road link segments, matching accuracy results were moderate only.

## 2.9 Map Matching Variable Sampling Rate Accuracy Evaluation

Test carried out by varying sampling rate of GPS point at different intervals showed that for probabilistic approach, 30 second time interval between two consecutive GPS points was good enough for getting overall accuracy of more than 80%.



**Figure 2.21: Map matching accuracy with varying sampling interval**

Accuracy evaluation as shown in Figure 2.21 shows the change in accuracy percent with change in sampling interval. Interval '0' represents the original data set which contains GPS point 3 or 5 second interval. Interval of 1-5 second was obtained by conducting simple linear interpolation on the original data set. However, linear interpolation result was not satisfactory. In the case for

sampling interval from 6-600 second, points in between the original data set were removed to maintain fixed interval. Accuracy percentage of 80% was maintained up until 30 second interval between consecutive GPS point. After 30 second, accuracy percentage dropped sharply at which only 16.9% was obtained when the interval was 600 seconds. The result could help make data collection from these probe vehicles more efficient as less amount of data could be collected with maintaining relatively high accuracy and reducing storage load.

### 2.10 Map Matching Speed Test Performance

Speed test performance was conducted in a ten-node distributed system. Each node is a Xeon(R) CPU with 16gb of memory. The total GPS probe data from the month of June and July 2015 is about 2.2 billion data rows.

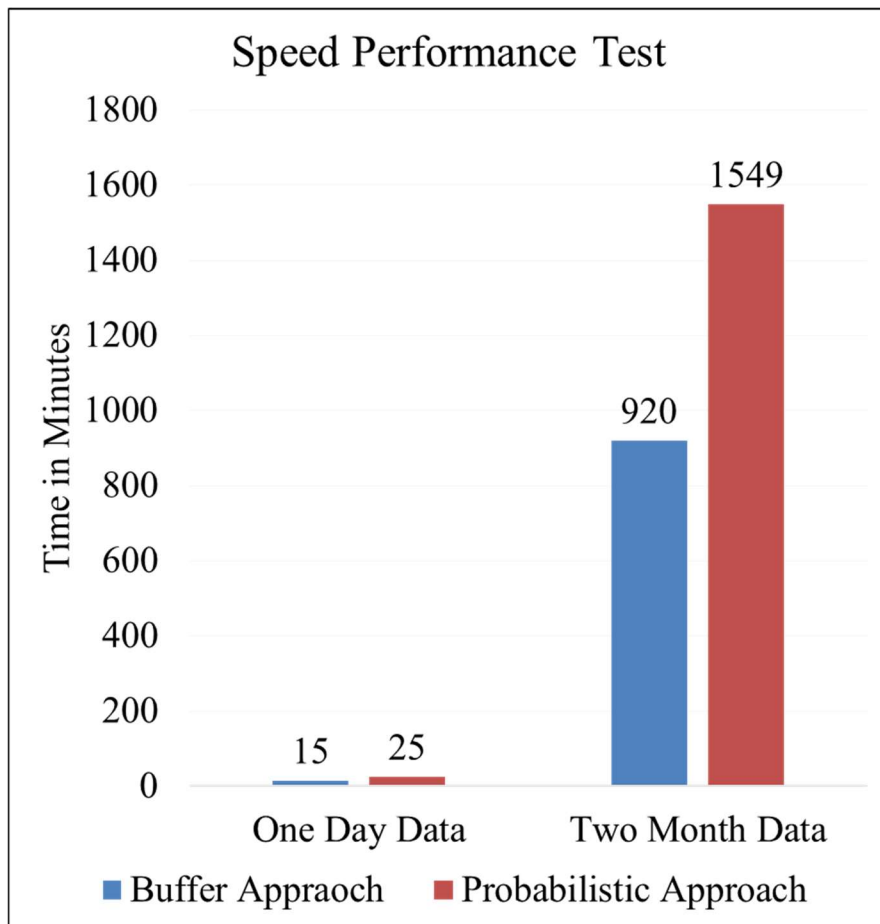


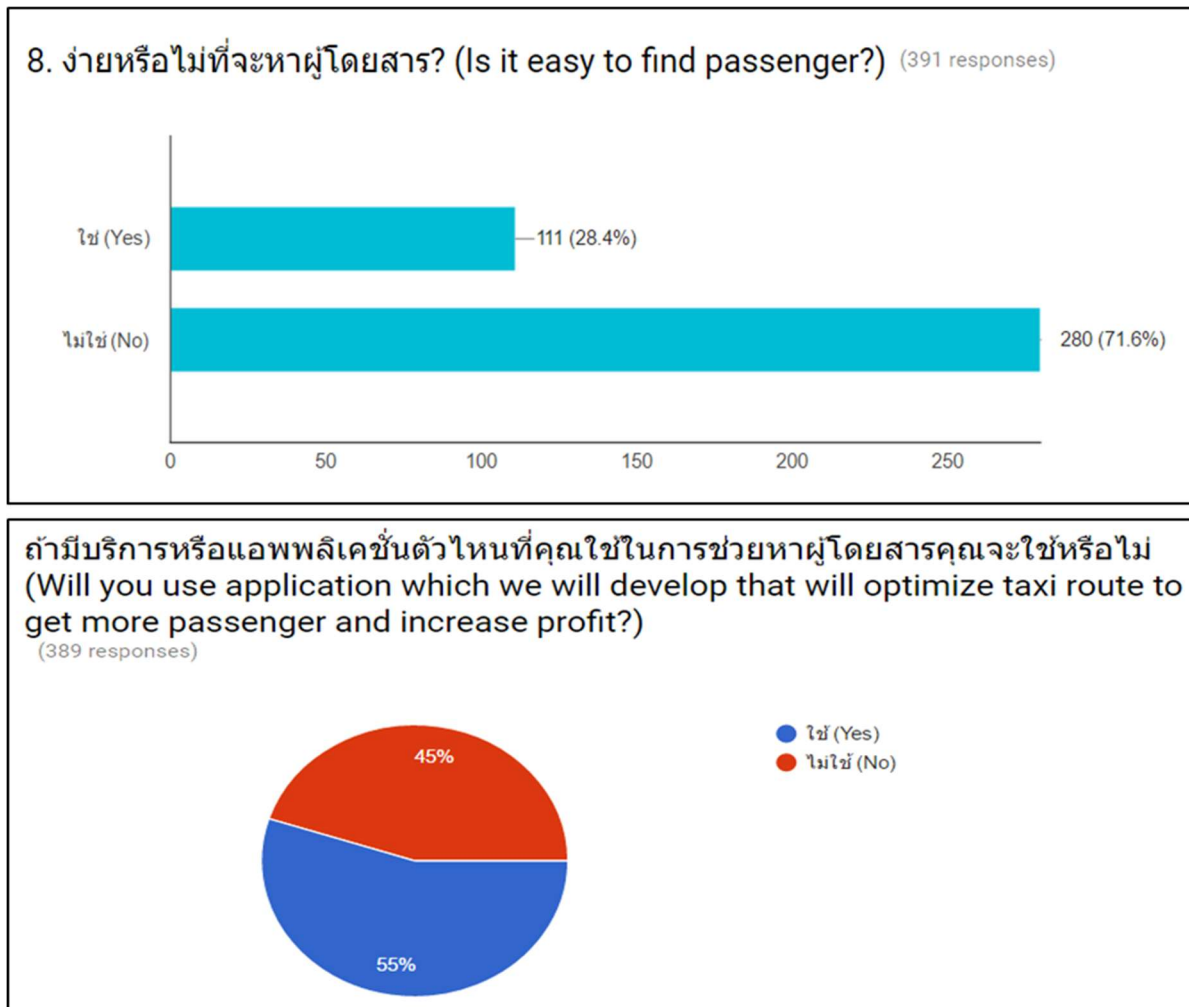
Figure 2.22: Speed performance test of map matching

Each data row consisted of a data specification as mentioned in Table 2.1. Since, buffer distance approach and probabilities approach produced significant accuracy result, a Hadoop/Hive user defined function script was created to process the whole two-month data set. As shown in Figure 2.22, probabilistic approach took about 1549 minutes to process two-month data with 25 minutes an average for one day data set. On the other hand, buffer distance approach took about 920 minutes for two-month data with 15 minutes an average for one day data set.

## **2.11 Bangkok Taxi Survey**

To assess the real situation of the taxi driver working, the real understand from the taxi driver itself is very necessary. Such understanding can be only obtained when a real survey was conducted among the drivers. Information from taxi driver about their issues and problem from their own experience could be vital when modeling drivers service. A questionnaire survey was conducted in the Bangkok in collaboration with the Asian Institute of Technology and Toyota Tsusho Electronic Nexty Thailand in the year 2016. The objective of the questionnaire survey was to understand the operation of taxi from the status of the taxi driver working perspective in Bangkok and surrounding provinces. A hypothesis was developed about Taxi operation by general understanding of the taxi operation. This survey when conducted provides the hypothesis to validate or invalidate, which then will be used as means for behavior simulation of taxi operation. The survey itself was conducted in three stages. In each stage, different number of taxis were surveyed as well as question were modified based on the reply received from the taxi driver. One important factor that needs to be considered while doing the questionnaire survey is that the drivers need to be inform prior regarding the incentive of questionnaire survey and how it could help benefit to their daily activity. In first stage of the questioner survey a total of 30 taxi drivers were questioned for 20 questions. The detail of the stage 1 questionnaire survey is shown in Appendix B The reply from the taxi drivers were assessed and modified for the next two stages of the questionnaire survey. The second stage taxi survey was conducted from 150 taxi drivers with 34 questions and the final stage of questionnaire survey was conducted from 400 taxi drivers with 34 questions as well. The detail of the stage 2-3 questionnaire survey is shown in Appendix C. The 2-3 stage of questionnaire survey is divided into four sections.

1. Section A: Personal Information
2. Section B: Taxi Working Information
3. Section C: Expenses
4. Section D: Application



**Figure 2.23: Bangkok taxi questionnaire survey**

The result from the survey revealed that there are issues related for the taxi operation based on taxi driver's perspective. One of the prominent issues was the amount of income they make as well as how frequently the driver can get the passengers as shown in Figure 2.23. More importantly, drivers were also interested in using the mobile application that would provide them with recommendation for finding the passengers. The detailed questionnaire response is presented in Appendix B and Appendix C.

## CHAPTER 3

### TAXI BEHAVIOUR SIMULATION

3.

#### 3.1 Introduction

Taxi behavior simulation model describes the behavior of taxi operation in the city level that shows how the taxi operates business as usual. Hence the question arises why understanding the taxi operation business as usual is important. As mentioned in Chapter 2, a Bangkok taxi survey was conducted with the questioner form. The objective of the survey was to understand the real situation of the taxi driver from the driver point of view. Out of many question asked, one of the question asked was whether taxi driver could get the passenger easily. Out of 400 taxi drivers asked the question almost 70% of the response was it is difficult to get the passenger. However, on the passenger side there are many complains from the taxi drivers of being rejected by the taxi drivers which is shown in Figure 3.1.

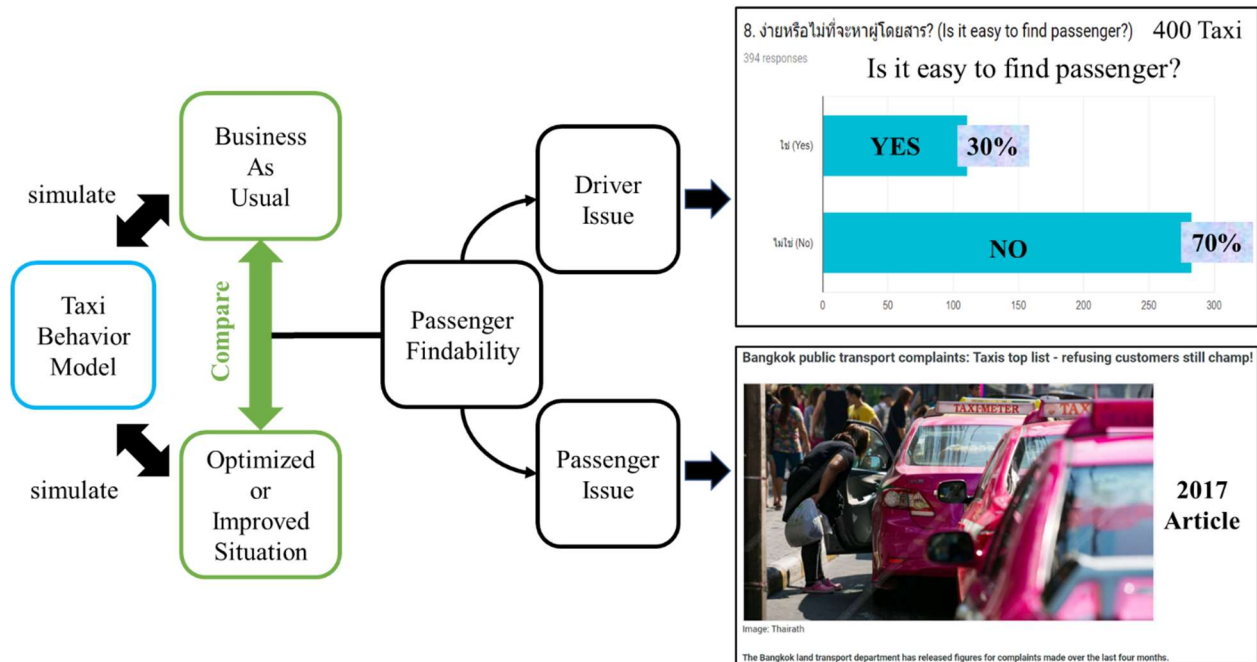
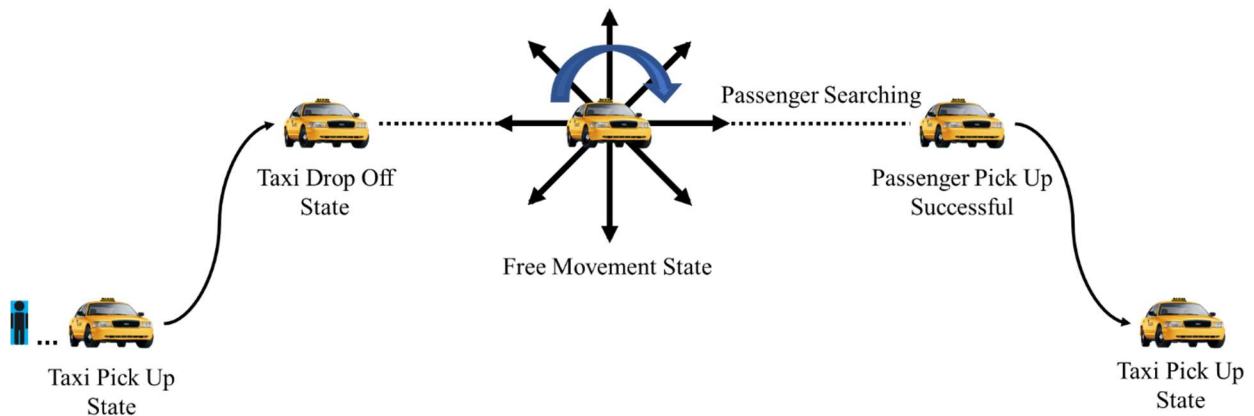


Figure 3.1: Need for taxi behavior simulation

Looking at the issue from both drivers' perspective as well as the passenger perspective, the issues is related with the passenger findability. Optimizing the taxi service operation could help minimizes the passenger findability issue. In order to make an optimization, understanding the business as usual is necessary. In this regard, business as usual could be obtained from simulating the taxi behavior on which improvement could be introduced to make the optimized taxi service.

### 3.2 Taxi Behavior Modeling State

Taxi behavior modeling consist of a multiple state. A simple state diagram for the taxi behavior is as shown in Figure 3.2. Moving from left to right side, when the taxi gets the passenger, the taxi state could be described by 'Taxi Pick Up State'. When the taxi has a passenger then the taxi has to go to the passenger destination place to drop off the passenger for which the state could be described as 'Taxi Drop Off State'. Following the passenger drop off, the taxis are free to move any direction to search the passenger which could be described as 'Free Movement State'. In the 'Free Movement State', taxi could move to 9 cardinal direction including staying at the same location. During this state taxi are searching for the passenger. As the taxi finally picks up the passenger then the taxi state is changed back to the 'Taxi Pick Up state. The state cycle continues until taxi stop for working.



**Figure 3.2: Taxi behavior modeling state**

Based on the taxi state diagram, taxi behavior is characterized by the dynamic discrete time dependent events involving customer pick-up, customer drop-off, cruising, and parking within the

spatial and temporal domain. Simulation models which are a simplification of a real-world system, can help understand effects of change of such dynamic behavior. In this regard, agent-based simulation and modeling, in which each taxi behaves as an agent, can capture such dynamic behavior through reconstructing complex patterns by decomposing complex systems down to the level of single agents that are administrated by sets of behavior rules (Baster et al. 2013). The advantage of agent-based modeling is that, rather than modeling the entire system with a single Equation, the entire system is modeled with the collection of autonomous taxi agent with rules governing them, which makes complex individual agent behave more naturally (Bonabeau 2002). In this way, agent-based simulation and modeling can highlight the effect of a change in taxi services and its impact to driver's income profitability through optimizing parameters (number of trips, passenger waiting time) derived from simulation. As an example, what will be the impact on taxi behavior service when the number of agents i.e., taxi is increased to the region of low taxi demand or decreased to the region of high taxi demand. Understanding such causality could help better management of taxi fleets with regards to the operational cost as well as improve taxi driver's income. Moreover, recently, many big cities, such as London and New York, have plans to adopt electric taxis (Tu et al. 2016), and understanding discrete taxi behavior through agent-based modeling could help optimize locations for charging stations, which are crucial for such electric vehicles.

As taxis services are operational throughout the city, spatial and temporal information from these vehicles can be an asset for governing different aspects of urban management. Information, as such, could contribute mainly to helping make better decision-making processes at both government and local level. Having said this, the constant advancement of collecting moving trajectory data in space and time has opened up the possibility of a wide range of study in the field of spatial information science (Sadahiro et al. 2013). One of the primary technologies for retrieval of information of various traffic data is from stationary equipment, like loop detectors, for automatic vehicle identification. However, they are limited to specific sections of road. On the contrary, a probe car, also known as a probe vehicle or floating car, utilizes the running vehicles to gather various traffic information, and has been an emerging ITS technology for modeling vehicle behavior (Bischoff et al. 2015; S. F. Cheng and Nguyen 2011; Miwa et al. 2004). Big cities, like New York and Beijing, have taxis already equipped with GPS sensors that collects spatial and



temporal data to a data center to be processed to extract traffic information (N. J. Yuan et al. 2013). The taxi driver mobility intelligence is an essential factor to maximize both profit and reliability within every possible scenario, and the knowledge about the service can be an advantage for the driver (Luis Moreira-Matias et al. 2013). However, to understand such stochastic dynamics of taxi behavior, micro-level simulation models are required, which can be further analyzed for optimization of taxi services by adjusting parameters like demand, supply, or altering dispatching algorithm (Maciejewski et al. 2016).

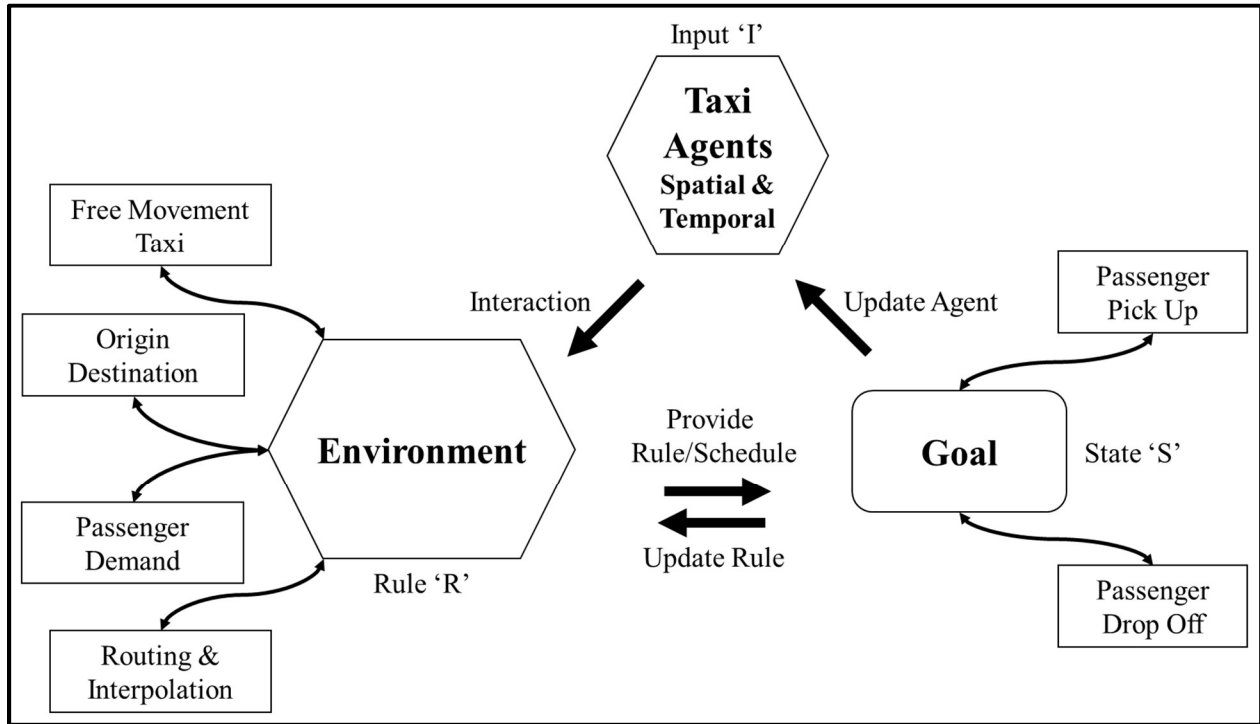
Agent-based modeling and simulation (ABMS) was implemented, to capture the dynamic behavior of taxi operation, for which in recent years, has been seen in many areas of application, such as flow evacuation, traffic, and customer flow management (Bonabeau 2002). Agent-based modeling and simulation describes the dynamic action of an entity i.e., taxi agent governed by behavior rule and properties, similar to the work presented in (Abar et al. 2017; S. F. Cheng and Nguyen 2011; Grau and Romeu 2015), to emulate the taxi behavior in Bangkok, Thailand.

The main task of the taxi simulation model is summarized as follows:

- Proposed a taxi agent regarding spatial and temporal domain based on a stay point cluster of probe GPS data and a kernel density of its timestamp.
- Formulated a concept of free taxi movement based on the movement direction of the taxi, which was introduced for searching passengers.
- Developed an agent-based simulation model which is based on multiple parameters (taxi stay point cluster; trip information (origin and destination); taxi demand information; free taxi movement and network travel time) that were derived from probe GPS taxi data. As such, agent's parameters were mapped into a grid network and the road network, for which the grid network was used as a base for query/search/retrieval of taxi agent's parameters, while the actual movement of taxi agents was on the road network, with routing and interpolation.

Agent Based Modeling (ABM), which works on the state-rule-input architecture (Torrens 2010), such that taxi behaves as an agent that interacts with the environment to capture the dynamic

behavior through reconstruction of complex pattern which is defined by the behavior rule. Figure 3.3 shows the conceptual design of the agent-based modelling.



**Figure 3.3: Conceptual design of agent-based modeling**

Conceptual design of agent-based modeling consists of an Input 'I', Rule 'R' and the State 'S'. The input of the agent-based modeling is defined by the taxi spatial and temporal variable. The agent spatial and temporal parameter is extracted from the stay point cluster from the probe GPS data. The agent interacts with an environment which provides the rule to the agent. The environment is defined by multiple variable including free movement taxi, taxi trip origin destination, taxi passenger demand and routing & interpolation for the network travel time. The rule then guides the agent to reach the goal which is defined by agent State. The agent is defined by two states. The first state is the taxi without passenger and the second state is the taxi with state. Depending upon the current state and the rule provided by the environment, next state is defined as the goal of an agent. Both input agent and environment are updated based on the goal reached. The updating of the environment provides the indirect interaction of an agent among each other. The motivation of taxi behavior simulation modeling is to optimize taxi service operation, through an increased number of passenger trips, making drivers wait a less amount of time to get their next

passenger, and making more extended passenger trips, as well as determine optimum working time based on the spatial and temporal domain. However, to identifying and evaluating such optimizing parameters, knowing real taxi behavior is a must. The proposed agent-based simulation and modeling recreates the real taxi behavior i.e. business as usual from which optimizing parameters could be derived, that would improve the taxi service for both driver, regarding monetary profit, as well as for passenger, regarding service level of the taxi.

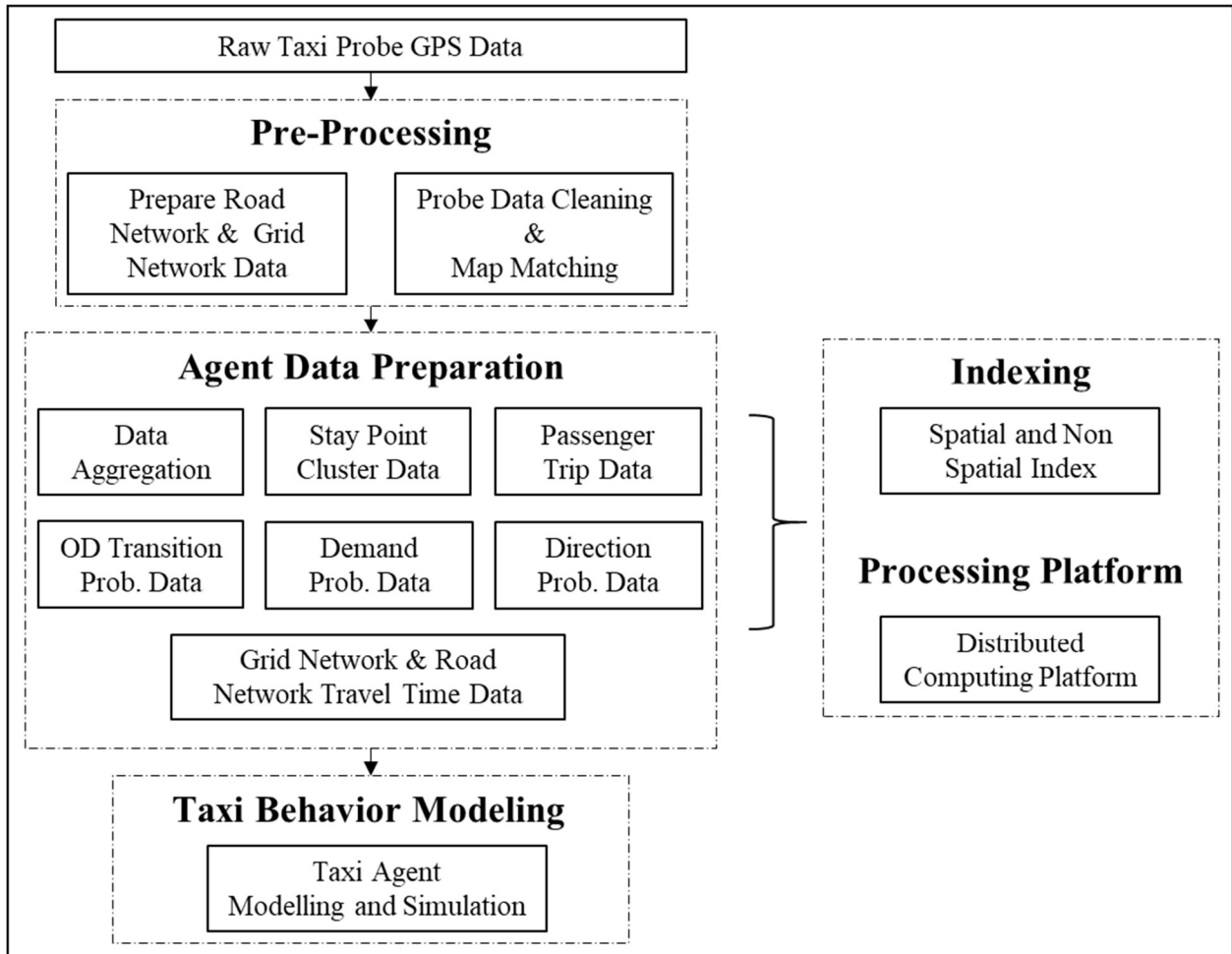
Also, spatial data, as such, probe GPS taxi data, with its ubiquitous properties, are enormous, and in most cases, deemed confidential. In such cases, obtaining raw spatial data is somewhat complicated. However, simulation and modeling techniques proposed in the study could essentially recreate such spatial data, with secondary data derived, and with properties as similar with the real data. In this regard, such simulation and modeling techniques are not only limited to vehicle behavior, but also could be implemented in simulating human mobility behavior from GPS or call detail record data.

### **3.3 System Overview**

The taxi simulation model itself is conducted with the agent-based simulation. However, before simulation operation is performed, data needs to be prepared and processed to extract the variable to be used for the simulation. As for this, the overall system is divided in three stages as shown in Figure 3.4. The three stages are as follows:

- Pre-processing stage
- Data preparation stage
- Taxi behavior modeling

In addition to the three stages that included preprocessing, data preparation and taxi behavior modeling, the overall processing is handled with spatial and non-spatial data indexing with distributed computing platform



**Figure 3.4: System overview**

In the data preparation stage, multiple secondary datasets from cleaned probe data were extracted, including stay point, passenger trip data, origin-destination probability data, demand probability data, direction probability data, and grid network road network travel time data. In taxi behavior modeling stage, agent-based modeling was implemented, that simulated the taxi behavior of the urban city.

Managing a large volume of data requires an efficient indexing technique that would handle index, search, and retrieval jobs (Chakka et al. 2003). Hence, both spatial and non-spatial indexing technique was implemented for the simulation purpose. STR tree, which is sort–tile–recursive R tree from Java Topological Suite (JTS), was implemented to index and search spatial data. As for

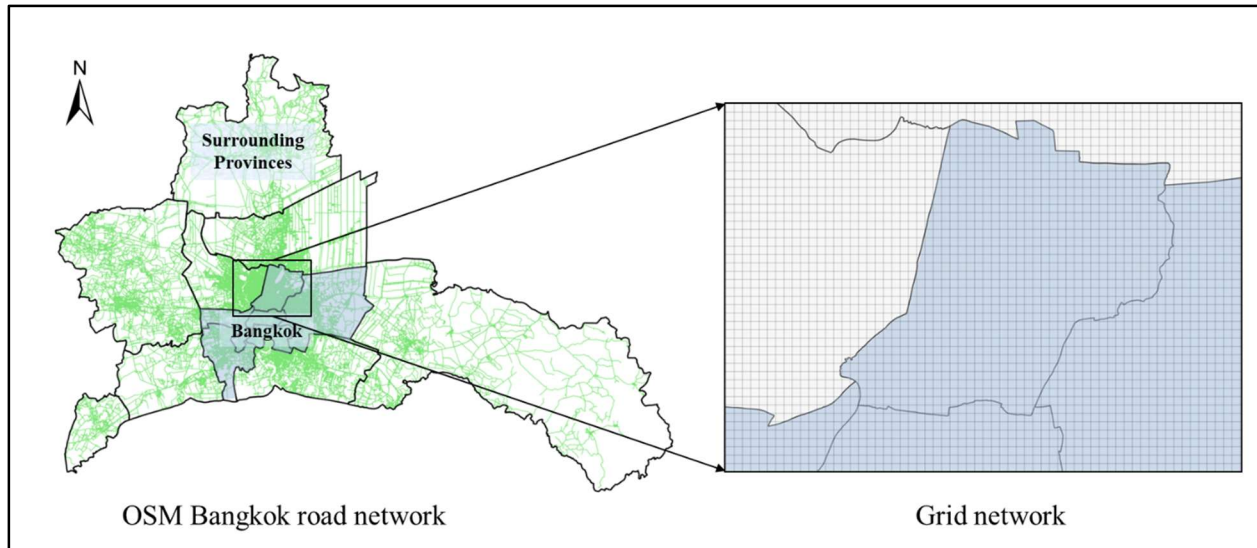
non-spatial data, an index and search engine named Lucene, that works on vector space model algorithms, was implemented for all query, search, and retrieval tasks during the simulating operation (Y. Zhang and Li 2009).

In addition to the large indexing volume of data, the preprocessing of all the data to be utilized for simulation, including cleaning, retrieving trip information, origin-destination, stay point extraction, direction movement extraction, was conducted in Apache Hadoop/Hive large-scale distributed computing system (Witayangkurn et al. 2013). The vehicle is the taxi that is running in and around Bangkok, Thailand, of which data is provided by Toyota Tsusho Nexty Electronics (Thailand) Co., Ltd, Bangkok, Thailand. The probe GPS data was collected from approximately 10,000 taxis with a sampling time of 3 or 5 second. The total GPS probe data preprocessed from 1 June 2015 to 31 July 2015 was about 2.2 billion data rows which were stored in Hadoop Distributed File System (HDFS). Each data row consisted of a GPS data points with data specification and sample data as described in Table 2.1. For spatial data processing, Apache Hive based query HiveQL (Hive Query Language) was developed including Hive UDF (User Defined Function) and Hive UDAF (User Defined Aggregated Function).

Each of the probe data collected belongs to the spatial trajectory generated by moving taxi in geographical space such that trajectory  $T_i = \{p_1, p_2, p_3, \dots, p_j\}$ , where  $p_j = (x_j, y_j, t_j)$ , such that  $x_j =$  longitude,  $y_j =$  latitude, and  $t_j =$  timestamp.

### **3.4 Road Network and Grid Network**

Open street map data of Thailand was utilized for the road network for which topological error was cleaned (Ranjit et al. 2017). Here, road network was represented by  $R$  such that  $R = \{r_1, r_2, r_3, r_4, \dots, r_n\}$ , where  $r_1, r_2, r_3, r_4, \dots, r_n$  is each road segment. The total of 228,416 OSM road network features was extracted for Bangkok and the surrounding provinces. Following the preparation of OSM road network data, taxi probe GPS data were preprocessed to remove erroneous datasets. The cleaned GPS data were then map-matched with probabilistic map-matching process, with open street map road network  $R$  (Ranjit et al. 2017), that mapped GPS data on the road segment which is described in Chapter 2.



**Figure 3.5: Left Open street map of Bangkok road network; Right Grid network**

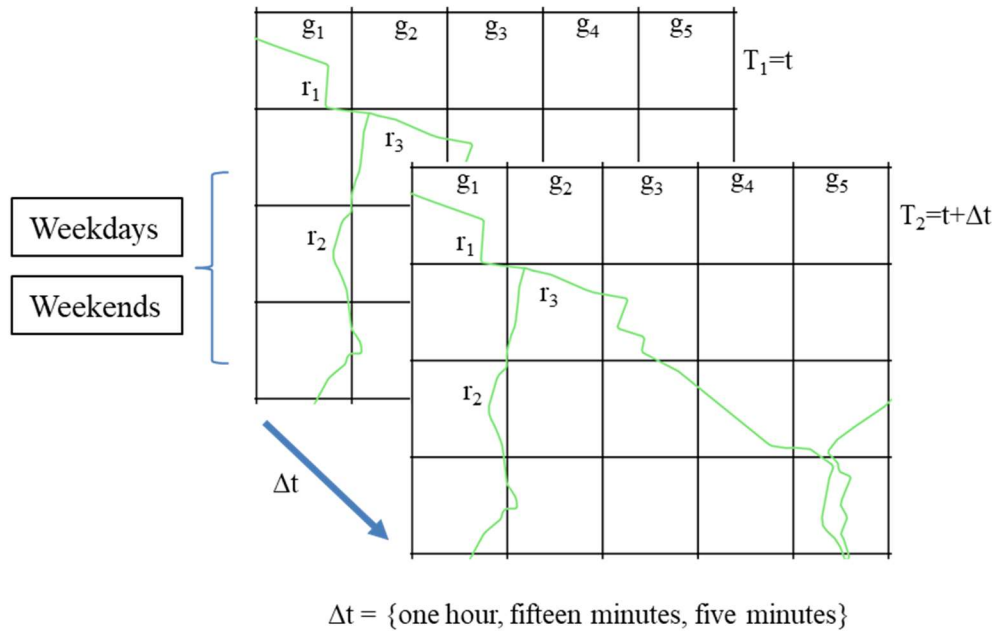
The small grid size of  $500 \times 500$  meters was chosen as grid network, in order to preserve spatial patterns and characteristics in the grid (Nam et al. 2016), however, the optimum grid size selection was still subjective, as larger grid size could be suitable for suburban or rural areas, but not suitable for dense urban area (Castro et al. 2013). A grid network of  $500 \times 500$  meters was constructed, covering all of Bangkok region, as well as surrounding provinces. Here, grid network was represented by  $G$  such that  $G = \{g_1, g_2, g_3, g_4, \dots, g_m\}$ , where  $g_1, g_2, g_3, g_4, \dots, g_m$  was each grid or cell. The total of 64,620 grid network features was prepared for Bangkok and the surrounding provinces. In addition to the road network map matching, the cleaned GPS data were also mapped to the grid network  $G$ . Figure 3.5 shows the OSM road network and grid network in Bangkok and surrounding provinces.

Grid network was used as it simplified the computation while maintaining both spatial and temporal relevance of the aggregated dataset. Also, use of grid network splits the given spatial region into disjoint areas, which makes it easy to inspect for further qualitative analysis (Castro et al. 2013). The road network was used during the preprocessing step of cleaning and map-matching probe taxi data, and then later used routing and interpolation of the simulated taxi agent trajectory.

### 3.5 Data Preparation

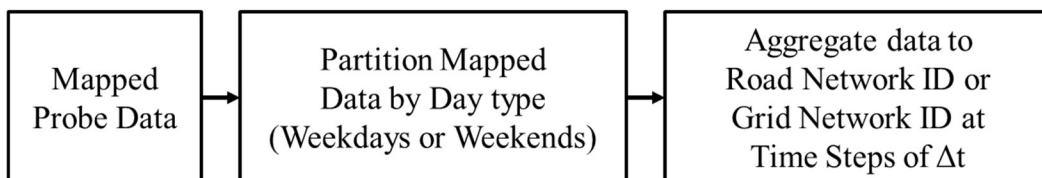
#### 3.5.1 Data Aggregation

Both road network and grid network mapped data, data were further categorized as “weekdays” and “weekends” data. Mapped and categorized probe data were then aggregated to each road segment and grid network at multiple time step of one hour, fifteen minutes and five minutes. Figure 3.6 shows road network “R” overlapped over grid network “G” at time steps of  $\Delta t$ .



**Figure 3.6: Probe data aggregation in OSM road network and grid network**

The flow diagram of a data aggregation process is shown in Figure 3.7. As mentioned, the probe data were at first mapped into the road network and the grid network. The mapped data were then partitioned based on weekdays and weekends.



**Figure 3.7: Flow diagram of a data aggregation process**

An aggregation function was used for aggregating the data for time steps of  $\Delta t$  at road network and grid network for both weekday and weekend data partitioned. Hence, the probe data as mentioned was categorized to weekday data and weekend data. In the period of two month of June and July 2015, weekday count for 45 days and weekend count for 16 days. Figure 3.8 shows the data aggregation example for every 1 hour interval. Each of the aggregated probe GPS data is indexed to the Grid Id for the given time interval. The aggregation as mentioned was also conducted for multiple time interval which were one hour, fifteen minutes and five minutes depending upon the type of processing required.

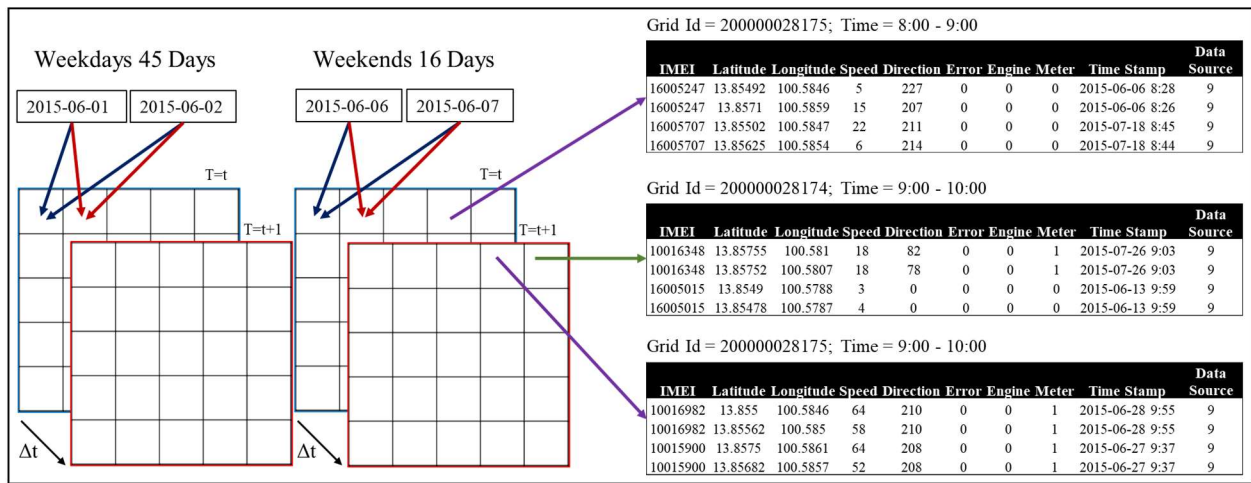


Figure 3.8: Data aggregation example

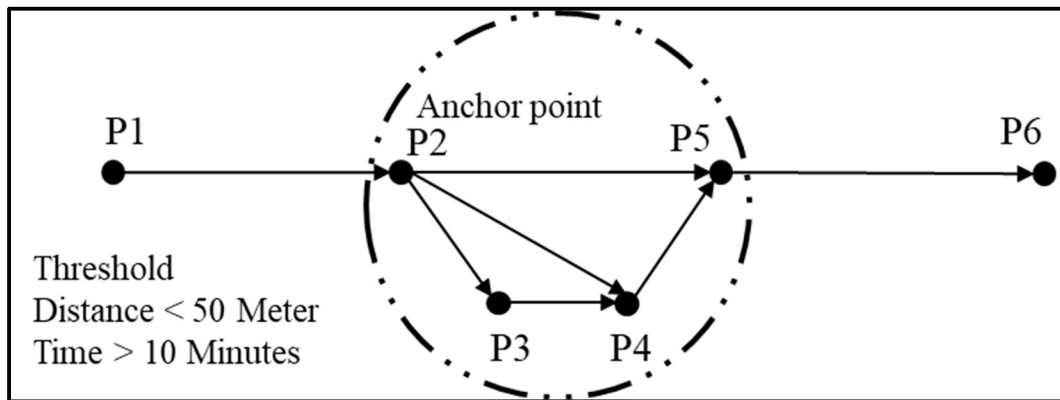
### 3.5.2 Stay Point Extraction

In order to conduct the taxi behavior simulation, the initial location and starting time of taxi agent needs to be defined prior. Initial location and starting time for each taxi agent was extracted based on the stay point cluster and kernel density of the cluster timestamp within the given spatial location. Stay point denotes locations where the vehicle (taxi) have stopped or stayed at some location for a certain interval of time, such as a parking place or a gas station, or while looking for a passenger. Taxi stay point cluster location was extracted, which depicts the start location for each taxi during the simulation. The stay point algorithm first checks the distance between the anchor point  $P_{anchor}$  P2, as shown in Figure 3.9, with the successor points  $P_{successor}$  (P3, P4, P5) within the distance threshold  $D_{threshold}$  value (Q. Li et al. 2008; Y. U. Zheng 2015).  $D_{threshold}$  was



chosen to be 50 meters, which was set empirically. Following this, the time span between anchor point and last successor point P5, which was within the distance threshold, was measured. If the time span measured was greater than the time threshold  $T_{threshold}$  value of 10 min, then stay point locations were detected for the given taxi, as shown in Equation (3.1).

$$Stay\ Point = \{(P_{anchor}, P_{successor}) < D_{threshold} \& (P_{anchor}, P_{successor}) > T_{threshold}\}..... (3.1)$$

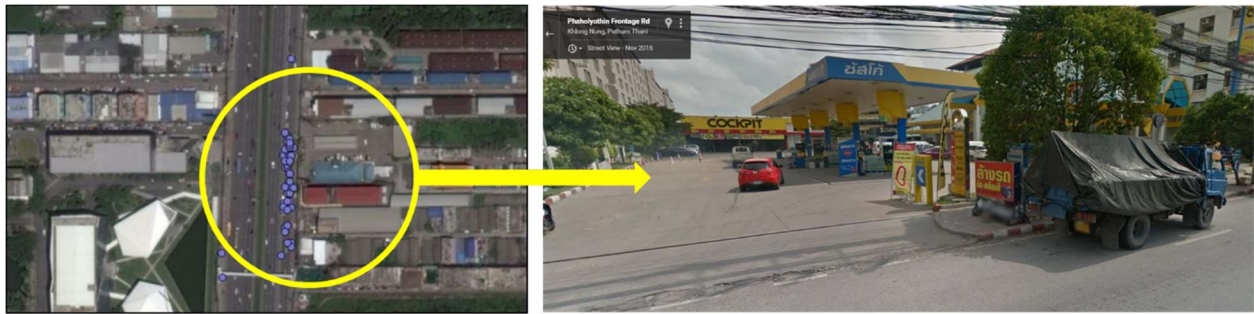


**Figure 3.9: Stay point extraction**

Stay points were extracted for the clustering as to simplify the processing for the clustering algorithm. As mentioned previously, the number of probe GPS data points from probe taxis was about 2.2 billion data rows. Clustering of this vast dataset would take lots of time as well as clusters would be difficult to separate as GPS traces would be dense in particular regions. To overcome this difficulty, stay point was extracted that would reduce the computational processing time and would provide meaningful clusters.

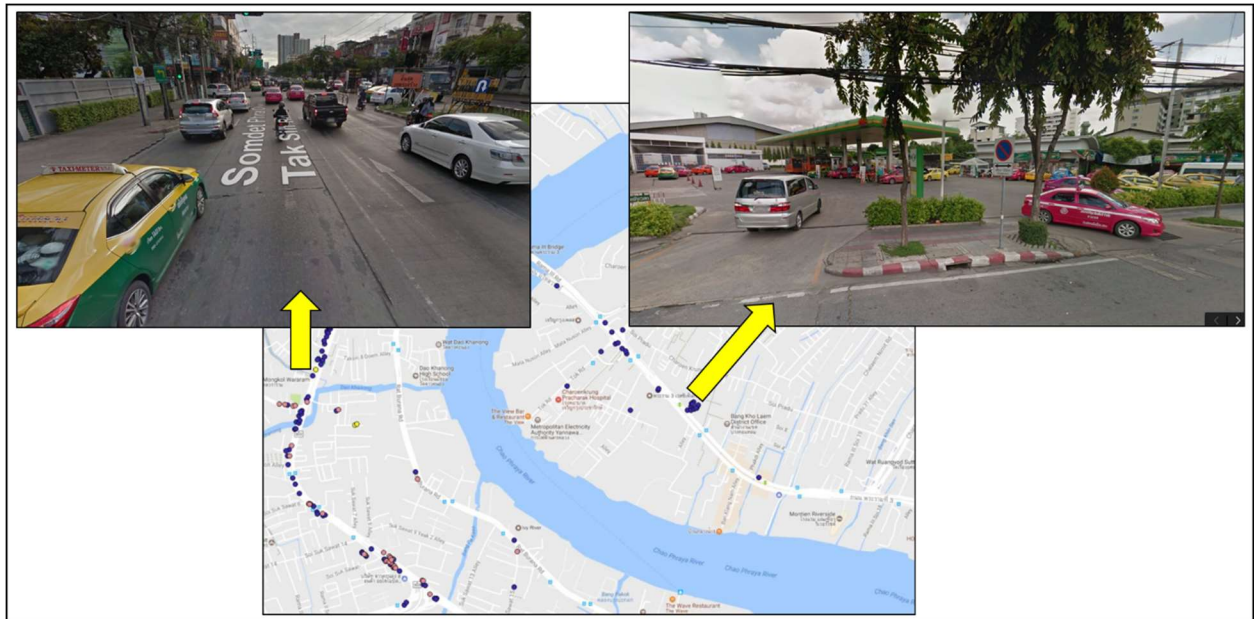
Next is the clustering of the stay point data extracted. Different clustering algorithm are available depending upon the type of clustering required. Some of the clustering algorithm which are used extensively are k-mean clustering (Kanungo et al. 2002; Macqueen 1967), mean shift clustering (Y. Cheng 1995; Comaniciu and Meer 1999), clustering using gaussian mixture models (GMM) (Verbeek et al. 2003), hierarchical clustering (Szymkowiak et al. 2001; Zhao et al. 2005), density based spatial clustering of applications with noise (DBSCAN) (Ester et al. 1996). Each of the clustering algorithm have their own pros and cons. As an example, k mean clustering can be easily implemented however it may not be suitable when the required number of cluster are known.

DBSCAN clustering are suitable for the spatial clustering however it may possess a challenge when the data is dense and computational time becomes high. Initially DBSCAN clustering algorithm was applied for the stay point data. DBSCAN produced the cluster from the stay point data that showed location where taxi normally stayed for longer period of time as shown in Figure 3.10.



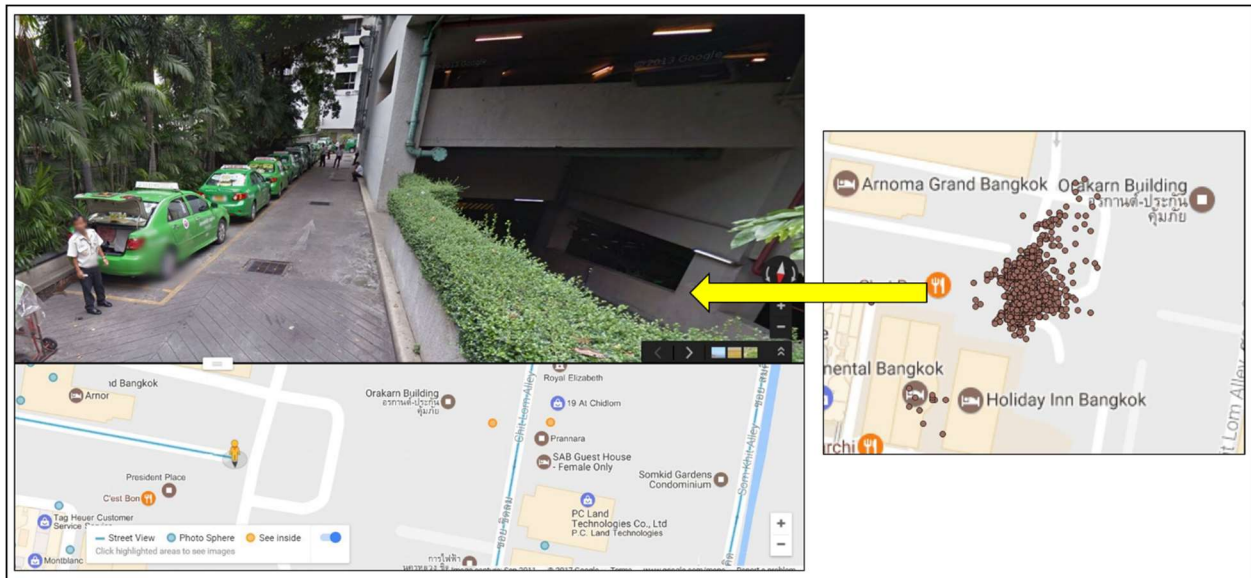
**Figure 3.10: Stay point cluster with DBSCAN algorithm**

However, normal DBSCAN algorithm produced some error cluster where some of the cluster did not seem to be a location where taxi could stay for long period of time. An example for the error cluster is shown in Figure 3.11.



**Figure 3.11: Example for the error cluster detected with DBSCAN**

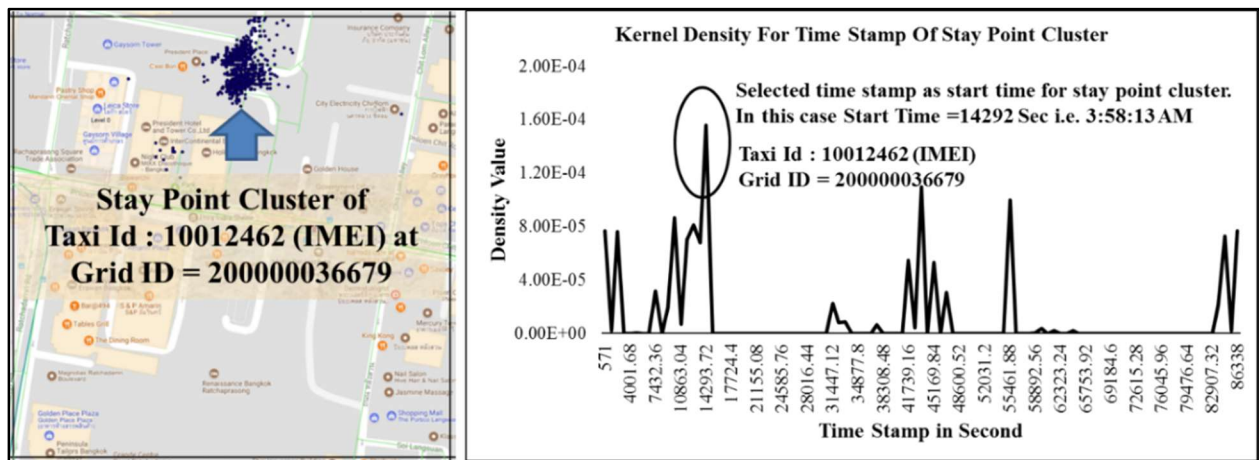
To overcome the error cluster, DBSCAN algorithm was improved by using grid based DBSCAN algorithm. A grid G based DBSCAN algorithm was implemented for each taxi stay point identified as proposed in (Ester et al. 1996; Gan and Tao 2015; D. W. S. Wong and Huang 2016), where the minimum number of points to form clusters (MinPts) and the maximum distance within two points that belongs to the cluster (epsilon  $\epsilon$ ) were chosen empirically, depending upon the number of clusters required. The MinPts value was chosen as 100 points, and epsilon distance  $\epsilon$  was chosen as 50 meters. However, both values could be adjusted depending upon the number of clusters required. Improvement on the clustering algorithm could be achieved with HDBSCAN clustering (Campello et al. 2013) which uses only single parameter, i.e., MinPts for the clustering algorithm. However, as stay points were extracted before clustering, with a threshold distance of 50 meter, all those points that were identified as stay points were within 50 meter threshold distance. Hence, traditional DBSCAN, with an epsilon distance of 50 meter, was used for clustering instead of HDBSCAN. Figure 3.12 shows the clustered stay point with the improved grid based DBSCAN method. Finally, the centroid of each cluster was computed that represented the stay location for each taxi.



**Figure 3.12: Stay point cluster with grid DBSCAN algorithm**

For each of the clusters, the start time was computed that represented the starting time for the taxi during simulation, using the kernel density function (Harvey and Oryshchenko 2012; Zucchini

2003) for the timestamp of each of the points in the clusters. High kernel density value of timestamp was chosen as the start time. The left Figure 3.13 shows an example for stay point cluster for taxi id “10012462” at grid id “200000036679”. The right Figure 3.13 shows the kernel density of timestamp all the stay point cluster for taxi id “10012462” at grid id “200000036679”. For this case, the density at timestamp “14292 seconds” (3:58:13 a.m.) was the highest, and hence, start time for this taxi was chosen at 3:58:13 a.m. and the cluster centroid as the starting location.

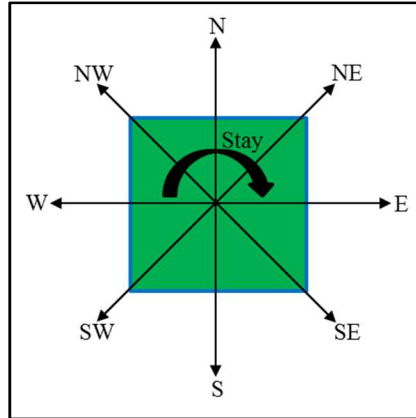


**Figure 3.13: Left: Stay point cluster; Right: Kernel density function of timestamp.**

### 3.5.3 Vacant Taxi Movement

The taxi state diagram shows that when the taxi drops the passenger to the destination, taxi changes its state to free movement state i.e. taxi with no passenger. When the taxi has no passenger, the taxi has a total of nine possible cardinal directions to move for searching the passenger, including north (337.5–22.5°), northeast (22.5–67.5°), east (67.5–112.5°), southeast (112.5–157.5°), south (157.5–202.5°), southwest (202.5–247.5°), west (247.5–292.5°), northwest (292.5–337.5°), or stay at the same location.

Figure 3.14 illustrated the nine possible cardinal directions for vacant taxi movement. Based on this assumption, vacant taxi movement was directed for searching of a passenger, with a directional probability which was estimated for both weekday and weekend data.

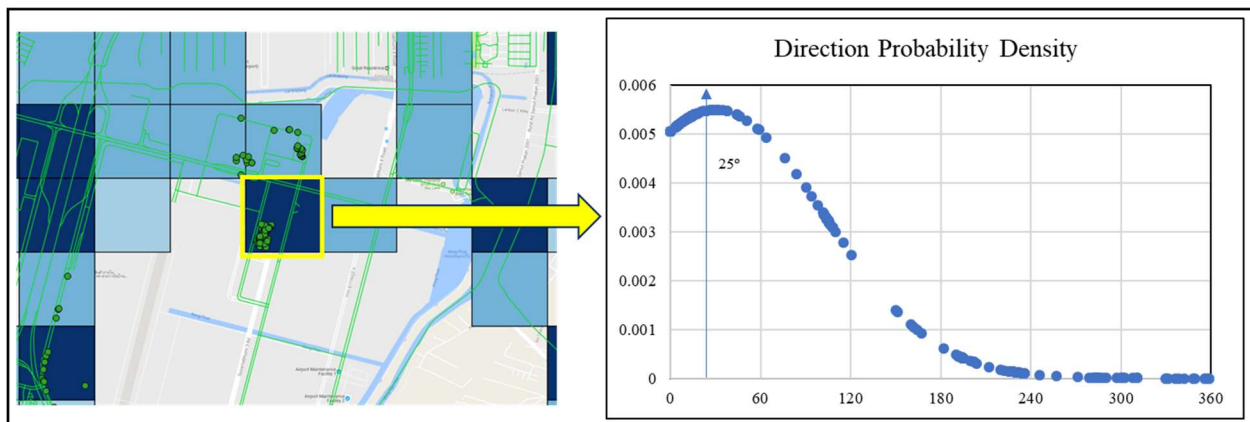


**Figure 3.14: Nine cardinal direction for vacant taxi movement**

The free movement of taxi was modeled based on the direction probability which is as shown in Equation (3.2), for each grid at a time interval of every 5 min.

$$\forall g \in G_t, P(d)_g = \frac{n_g}{N_g} \dots\dots\dots (3.2)$$

where  $P(d)_g$  in Equation (3.2) is the direction probability for vacant taxi movement, moving to direction  $d$ , for all grid  $g \in G$  at time interval  $t$ , such that  $n_g$  and  $N_g$  are the number of vacant taxi points moving to direction  $d$ , and the total number of vacant taxi points in grid  $g \in G$ , and time interval of  $t$ , respectively. The direction the vacant taxi would choose for searching for passengers was estimated from the highest  $P(d)_g$  obtained for a given grid and time interval.

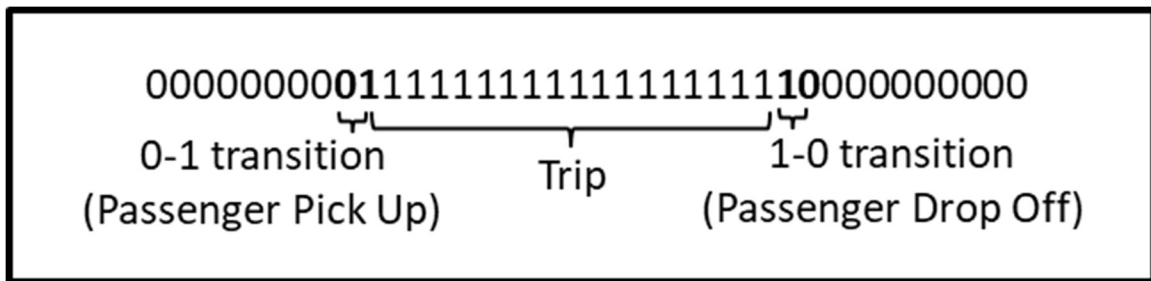


**Figure 3.15: Direction probability density of in a grid at given time interval**

An example of a directional probability density at a given time interval is shown in Figure 3.15. Based on the density at this selected grid at a given time interval taxi is more likely to move to northeast direction with directional value of 25 degree to search for the passenger. For each grid at the five minutes time interval directional probability is pre-computed which was used for the free movement taxi during agent-based simulation. Though the movement of taxi was from one grid to another grid, the actual movement was based on the OSM road network during the taxi agent simulation process.

### 3.5.4 Taxi Origin and Destination

In the taxi pick up state, the taxi simply need to go to the passenger destination. To model the passenger destination, origin and destination of the passenger trip from the taxi probe data need to be extracted and evaluated. Taxi origin and destination or simply OD refer to the location where the taxi picked up and dropped off the passenger or customer (Gonzales et al. 2014). The OD of the taxi was extracted from the taxi trip information, which is tracked through taxi “meter status” of probe data, as described in Table 2.1.

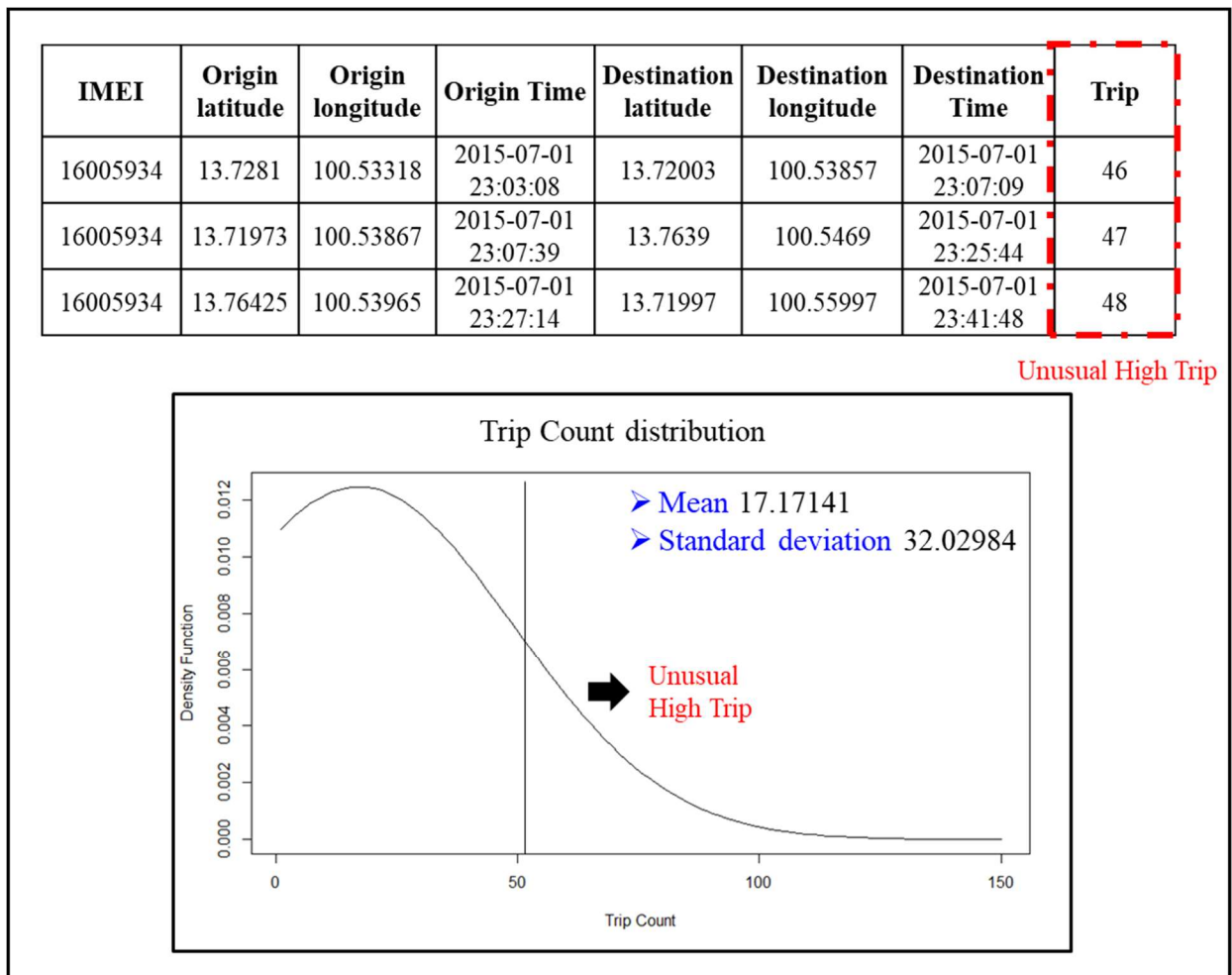


**Figure 3.16: Passenger trip based on pick-up and drop-off transition**

As for the trip itself, only those trips were considered whose origin and destination was within Bangkok and the surrounding provinces, as shown in Figure 3.5. This also suggested that longer trip information was eliminated, and simulation focused more on Bangkok region. Figure 3.16 shows the passenger trip based on pick-up and drop-off transition. Meter status 0 indicated that the taxi has no passenger and meter status 1 indicated that the taxi has passenger. Meter status

transition from 0 to 1 was considered as the start of a passenger trip, and transition from 1 to 0 was considered as the start of the non-passenger trip.

One issue when estimation the trip from the probe taxi data was some of the taxi showed usual high number of trip for a given day. As shown in Figure 3.17, one of the taxi with IMEI value of 16005934 has number of trip to 48 in one day. However, such high number of trip are uncommon in Bangkok which was also verified from the questionnaire survey from the taxi driver as shown in Figure 3.18. According to the questionnaire survey conducted among 400 taxi drivers in Bangkok and surrounding region almost 50% of the driver respond as that they make about 10 to 15 trips per day.



**Figure 3.17: Unusual high number of trip in certain taxi**

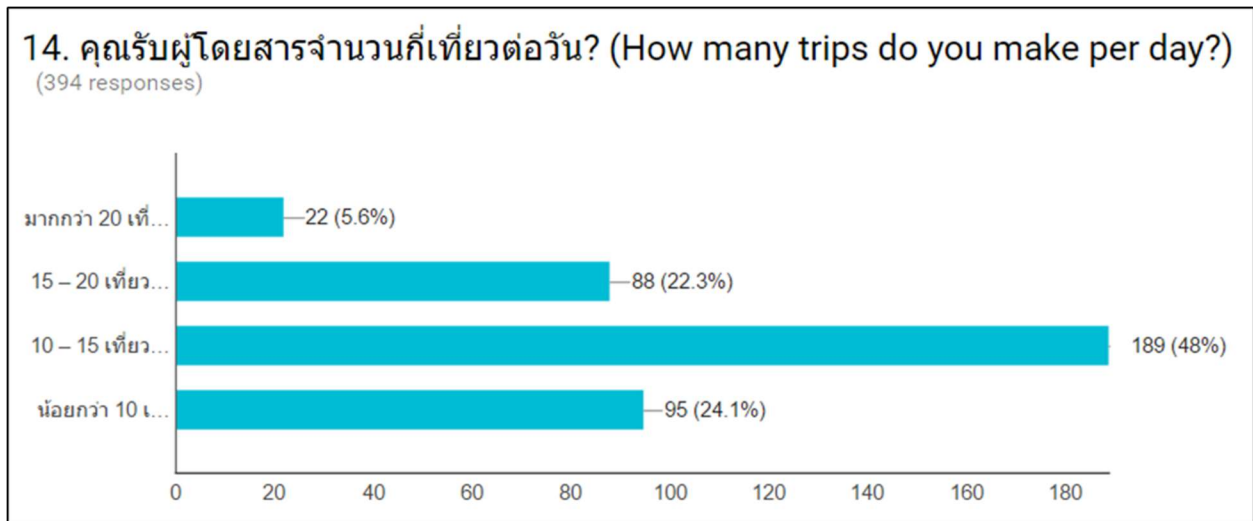


Figure 3.18: Number of taxi trip according to taxi questionnaire survey

One reason of this unusual high number of trip could be the error in the GPS probe data itself. Upon conducting the meticulous check up on the probe data, it is found that the some of the taxi data had error with frequent meter status being change from 0 to 1 and 1 to 0.

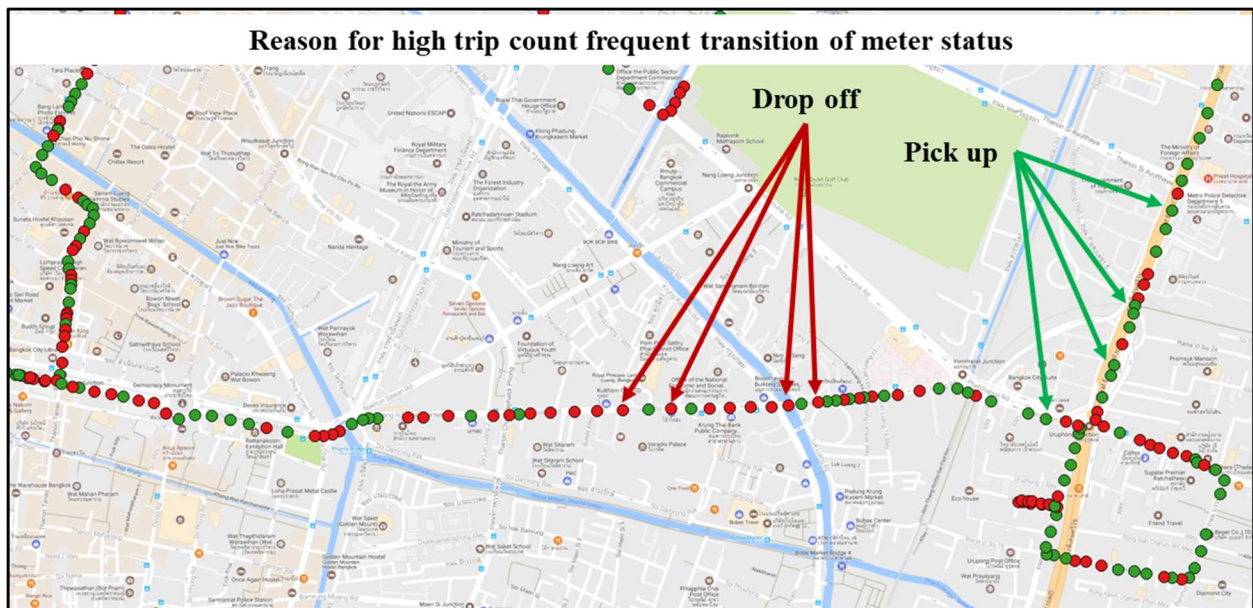
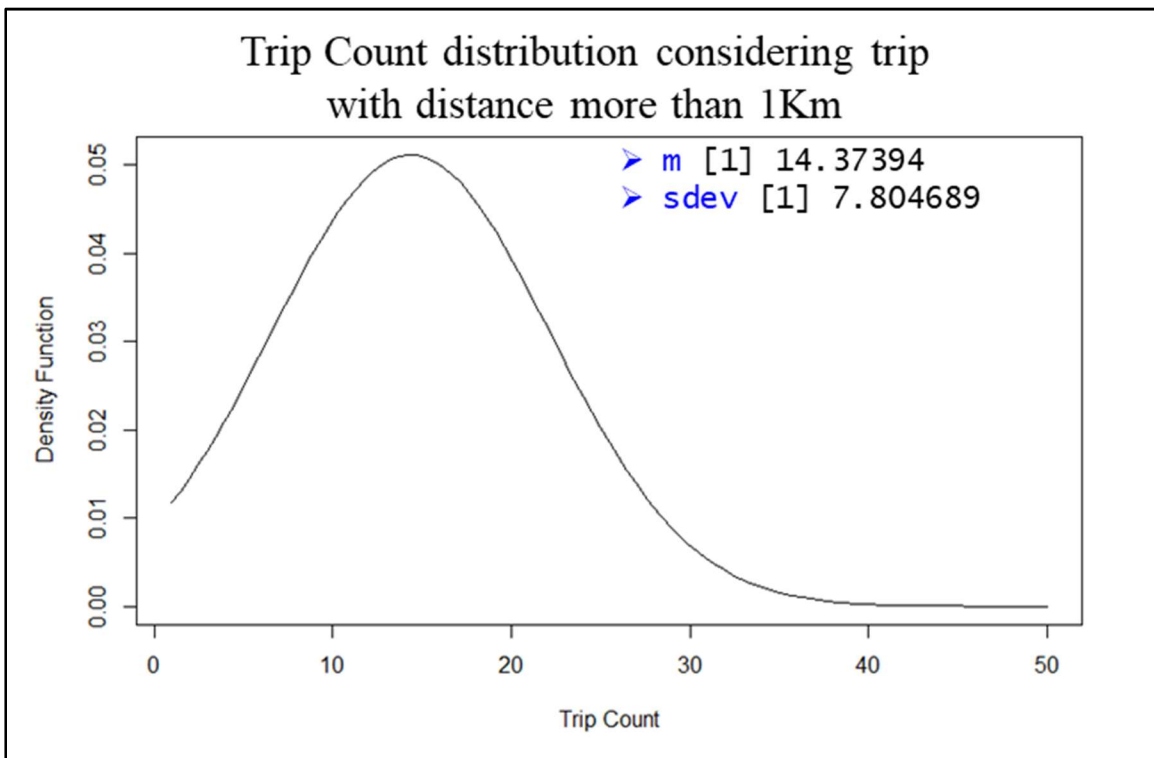


Figure 3.19: Frequent transition between pickup and drop off

Figure 3.19 shows an example of a taxi trip with unusual high trip being made for one of the taxi. When there is passenger location is shown in green color and where there is no passenger location



is shown in red color. As shown it is clear why some the taxi had high number of trip. The meter status has been constantly changing from 1 to 0 and 0 to 1 for the short distance covered and time travel. In some cases, there is continuous change from 0 to 1, 1 to 0 and 0 to 1. As the algorithm detects this transition, each transition is considered as start of the trip or end of the trip. A simple method was implemented to address this issue based on trip distance. Hence, only those trips were considered whose trip distance was more than one kilometer. Considering this criterion even though frequent transition is prevalent they were not considered as taxi trip. Figure 3.20 shows the trip distribution when trip with more than 1 km distance was considered. As compared to the trip without considering minimum threshold distance of 1 km in Figure 3.17, the trip with 1 km threshold distance adjusted the mean trip considerable similar with the real situation as suggested by the questionnaire survey as well.



**Figure 3.20: Trip distribution considering trip more than 1 Km**

With trip data cleaned each of the transition locations were mapped onto the grid network  $G$ , along with trip distance and trip time computed. A total of 3,081,230 passenger trips and a total of

2,570,432 non-passenger trips were recorded for June and July 2015. Table 3.1 shows the passenger and non-passenger trip examples for taxi id: 10012462 on weekday.

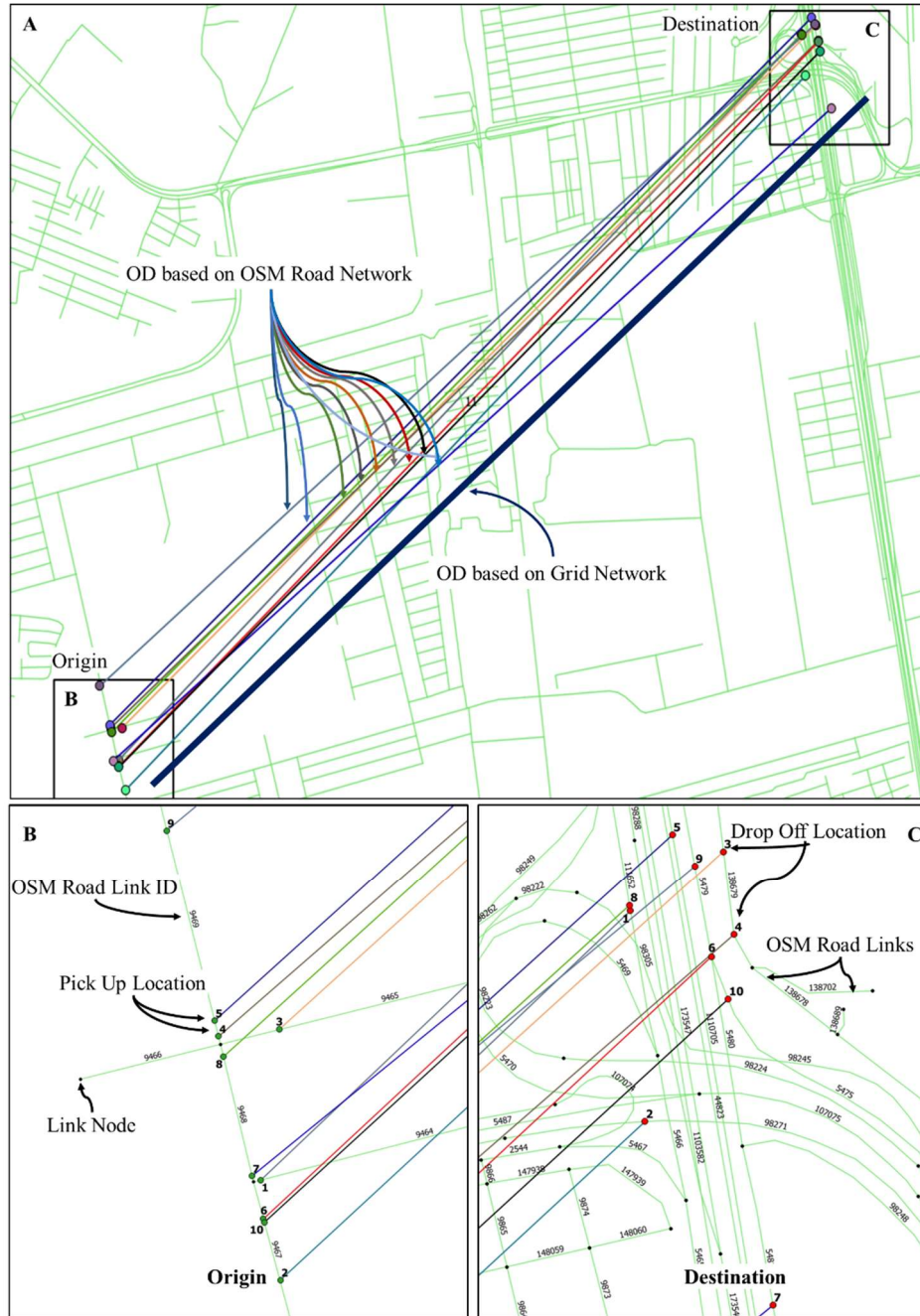
**Table 3.1: Passenger and non-passenger trip**

<b>Date: 2015-06-01; Day Type: Weekdays</b>						
<b>Passenger Trip</b>						
<b>IMEI</b>	<b>Pick Up Grid</b>	<b>Drop Off Grid</b>	<b>Pick Up Time</b>	<b>Drop off Time</b>	<b>Trip Distance (Km)</b>	<b>Trip Time (Minutes)</b>
10012462	200000035920	200000036681	9:04:56	9:09:57	2.34	5.02
10012462	200000036680	200000040994	9:22:01	15:44:16	62.35	382.25
<b>Non - Passenger Trip</b>						
<b>IMEI</b>	<b>Drop Off Grid</b>	<b>Pick Up Grid</b>	<b>Drop off Time</b>	<b>Pick Up Time</b>	<b>Trip Distance (Km)</b>	<b>Trip Time (Minutes)</b>
10012462	200000036681	200000036680	9:09:57	9:22:01	2.95	12.07
10012462	200000040994	200000037087	15:44:16	16:35:59	17.63	51.72

Based on passenger trip data, an Origin Destination or simply OD matrix was established for a time step of 1 hour, for which pick-up location (origin) and drop-off location (destination) pairs were aggregated, based on each grid network  $G = \{g_1, g_2, g_3, g_4, \dots, g_m\}$ .

As mentioned previously, there were about 30 million passenger trips recorded for a 2-month period, hence there are about 500,000 OD matrixes derived for each individual day, approximately. The OD matrix of this scale could be easily matched to the closest OSM road network. Doing so, however, generated issues where many OD pairs became sparse, with only one transition between origin and destination road network segment at the given time interval, as shown in Figure 3.21. Figure 3.21A shows the case when the OD became sparse, with only one transition between origin road segment and destination road segment, when created using OSM road networks as compared to when created with the grid network, which had 10 transitions between origin grid and destination grid, at the given time interval. Figure 3.21B shows the pick-up location at the origin with respect

to the OSM road network. Similarly, Figure 3.21C shows the drop-off locations at the destination with respect to the OSM road network.



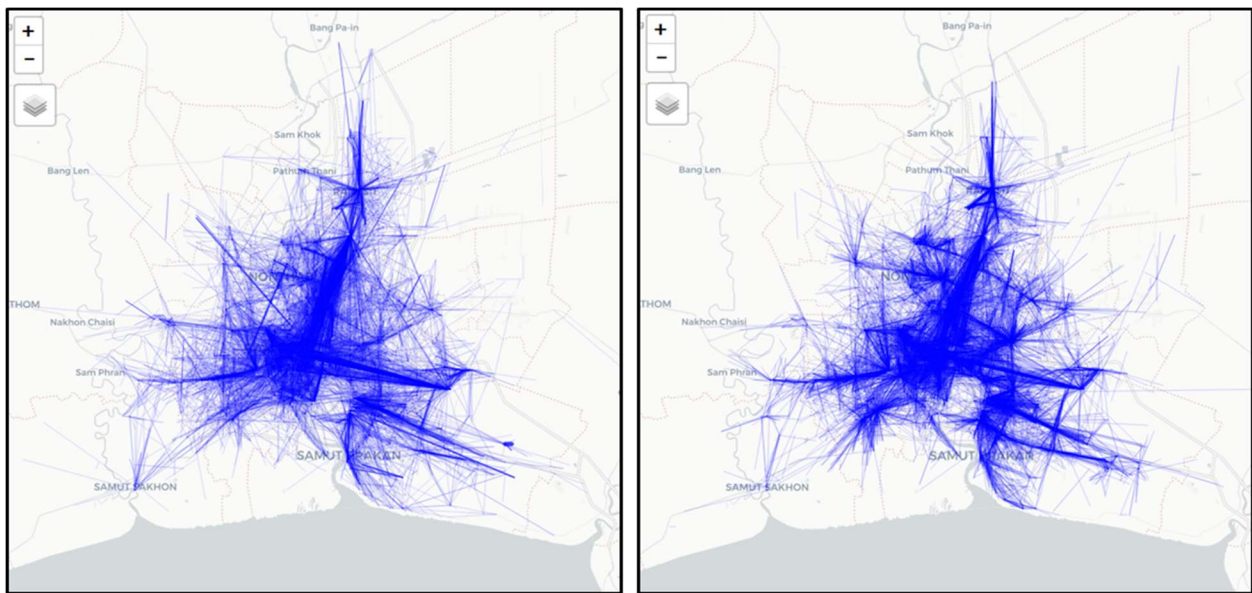
**Figure 3.21: (A) OSM road network and grid network OD comparison; (B) Pick up location at origin with respect to OSM road network; (C) Drop off location at destination with respect to OSM road network**

The Figure 3.21 shows each OD pair corresponding to road network were at different segments, hence creating the sparseness in the OD matrix. Using grid network helped reduce the sparsity problem when creating the OD matrix. In the case when a longer period dataset or larger numbers of taxi data were available, using road network becomes a better option than using the grid network. Other advantages of an OD matrix based in grid was that it could help anonymize the data where data are sensitive, and privacy needs to be protected as well as; such an OD matrix could be applied for understanding interzonal or intercity mobility as well (Ge and Fukuda 2016).

With the origin-destination matrix for passenger trip developed, OD transition probability was computed for both weekday and weekend data as shown in Equation (3.3), which was to be used for passenger trip simulation as.

$$\forall g \in G_t, P(g_{O \rightarrow D}) = \frac{Trip_{O \rightarrow D}}{Trip_O} \dots\dots\dots (3.3)$$

where  $P(g_{O \rightarrow D})$  is the OD probability for all grids  $g$  that belongs to  $G$  at time interval  $t$ , such that  $Trip_{O \rightarrow D}$  is the total number of passenger trips between the origin grid  $O$  and the destination grid  $D$ , and  $Trip_O$  is all the passenger trips that originated at grid  $O$  at time interval  $t$ .



**Figure 3.22: Passenger trip OD at time interval. Left: 7-8 a.m.; Right: 18-19 p.m.**

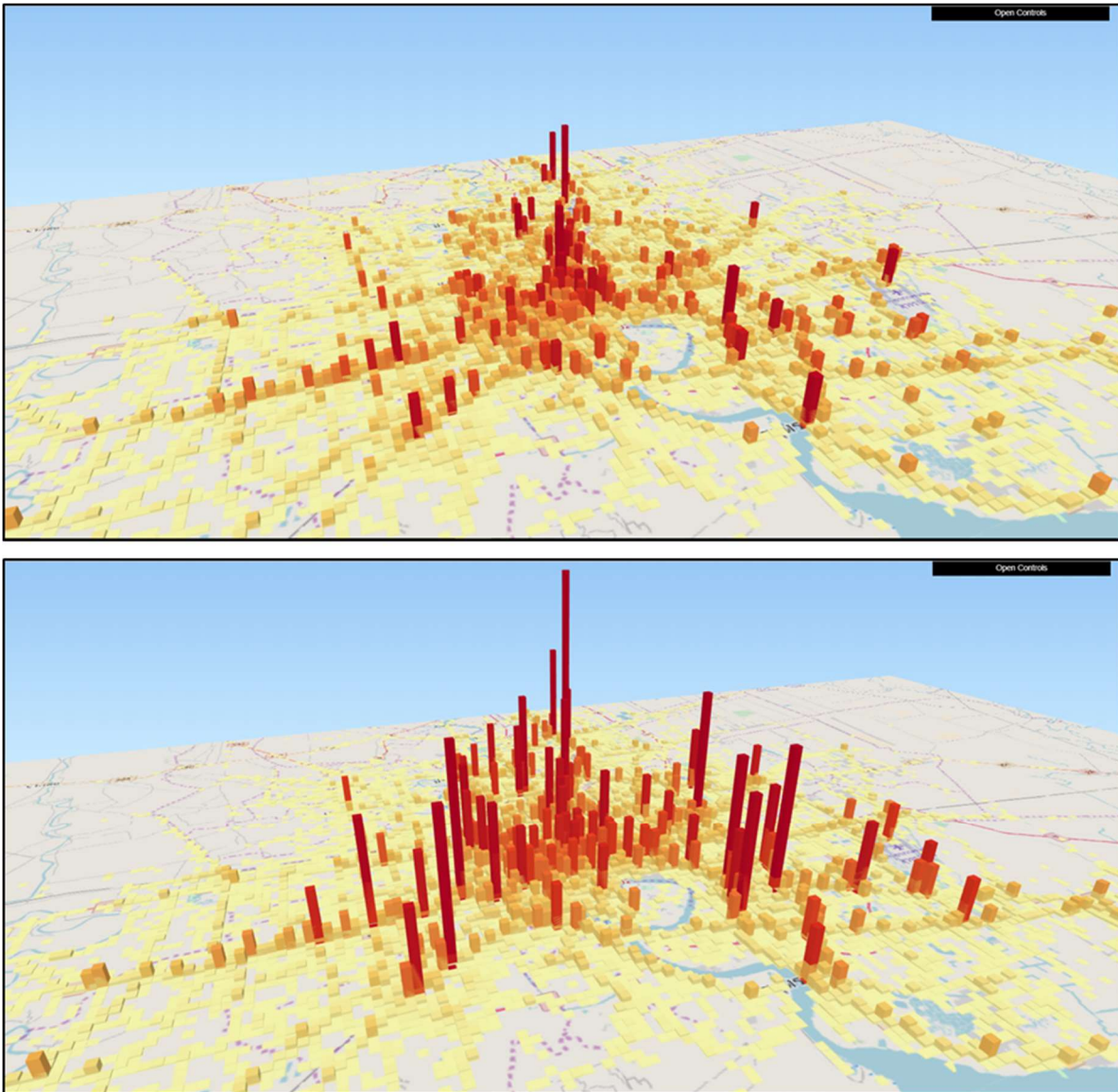
The concept behind the use of conditional probability of the OD matrix as shown in Equation (3.3) was to construct the transition probability matrix between origin and the potential destinations, such that for any given passenger trip generated during simulation, taxis at the origin would move to the destination estimated by the OD transition probability matrix for the given time interval. The OD probability transition matrix, however, could be the subject of periodic update (Luís Moreira-Matias et al. 2016) at specific time intervals as agents start to move along the spatiotemporal domain, which could provide the indirect interaction between the agents during the simulation process. Figure 3.22 shows the passenger trip OD visualization at a time interval of 7 a.m. to 8 a.m., and 18 p.m. to 19 p.m. Each line segment in the OD visualization represented the passenger trip from origin grid  $O$  to destination grid  $D$ , while the width of the line segment represented the number of trips. The higher the number of trips, the broader was the line segment, and vice versa.

### 3.5.5 Taxi Demand

In the taxi modeling state of free movement state, taxis are constantly looking for the passenger by moving to nine cardinal direction including the stay at same location. During the passenger searching process whether the taxi would get the passenger was defined by the available demand probability of success for a given spatial location and time interval. Passenger demand for the given time interval was defined as the total number of passenger pick-ups within the given spatial region at a given time interval. In general, demand is the passenger pick-up count in the spatial and temporal domain (J. Ke et al. 2017; R. C. P. Wong et al. 2014; J. Yuan et al. 2011; D. Zhang et al. 2016). Demand for a taxi was computed with the concept of probability of success or probability of finding passenger, as proposed by (Lv et al. 2017a; R. C. P. Wong et al. 2014), which was the demand that generated in the grid over the number of vacant taxis in that grid in that given time interval for both weekday and weekend data, as shown in Equation (3.4).

$$\forall g \in G_t, P(dm)_g = \frac{o_g}{v_g} \dots\dots\dots (3.4)$$

where  $P(dm)_g$  is the probability of success for all grid  $g \in G$  at time interval  $t$ , such that  $O_g$  and  $V_g$  are the total number of demands generated and total number of recorded vacant taxis at grid  $g \in G$  and time interval  $t$ , respectively. The time interval for demand probability was chosen for every 1 hour interval. Figure 3.23 shows the aggregated demand of taxi in morning hour from 7 a.m. to 8 a.m. and in the evening hour from 18 p.m. to 19 p.m.



**Figure 3.23: Aggregated taxi demand at time interval. Top: 7-8 a.m.; Bottom 18-19 p.m.**

The passenger demand for each grid and each time interval varies as shown in Figure 3.23, which also implies that the probability of success varies accordingly. This indicated that the probability

of success was subject to spatial and temporal variation, which could be captured through demand estimation. However, different levels of demand-related information (such as conservative, empirical, informed in real time, and informed about predictions) using data-driven Artificial Intelligence (AI) technologies, and how the information is shared among the driver, could alter and improve the overall behavior and needs to be defined carefully for better demand estimation (Grau et al. 2018).

### 3.5.6 Network Travel Time

Modeling of the taxi behavior is subjected to the movement of taxi from passenger pick up location to passenger drop off location as well as free movement while searching the passenger. Both movement are related on how taxi travel between the location along with travel time in the road network. GPS probe data now enables us to collect the travel time information through the moving vehicle, and has the given convenience to make travel time predictions in a complicated road network, whereby road network travel time can be estimated by estimating the speed between the nodes of the road segment (Liu et al. 2006, 2007). GPS probe data can essentially provide information of a traffic condition of a given period, such as travel time estimation, as well as traffic congestion, which directly relates to the distance travelled by a vehicle in that period (Wang et al. 2011). The average road network segment speed and average grid network speed was estimated for the time step of every 15 min time interval for both weekday and weekend data, as shown in Equations (3.5) and (3.6). The estimated average road network segment speed and average grid network speed was used for estimating taxi travel time during the simulation movement.

$$\forall r \in R_t, \bar{s}_r = \frac{\sum S_{p \in r}}{N_{p \in r}} \dots\dots\dots (3.5)$$

$$\forall g \in G_t, \bar{s}_g = \frac{\sum S_{p \in g}}{N_{p \in g}} \dots\dots\dots (3.6)$$

where,  $\bar{s}_r$  and  $\bar{s}_g$  are the average speed on the road network segment  $r \in R$ , and grid network  $g \in G$ , respectively at a time interval of  $t$ , such that  $\sum S_{p \in r}$  and  $\sum S_{p \in g}$  are the sum of the speed of all the points  $p$  with  $N_{p \in r}$  and  $N_{p \in g}$  as the total number of points that belonging to its respective

network. The use of two different average speeds was because not all road network segments had the probe GPS point associated with it at a given time interval. In such cases, the road network segment average speed could not be computed. Hence, grid network average speed was used instead for the given road network segment associated with it at the given time interval. In this regard, road network travel time was given by Equation (3.7).

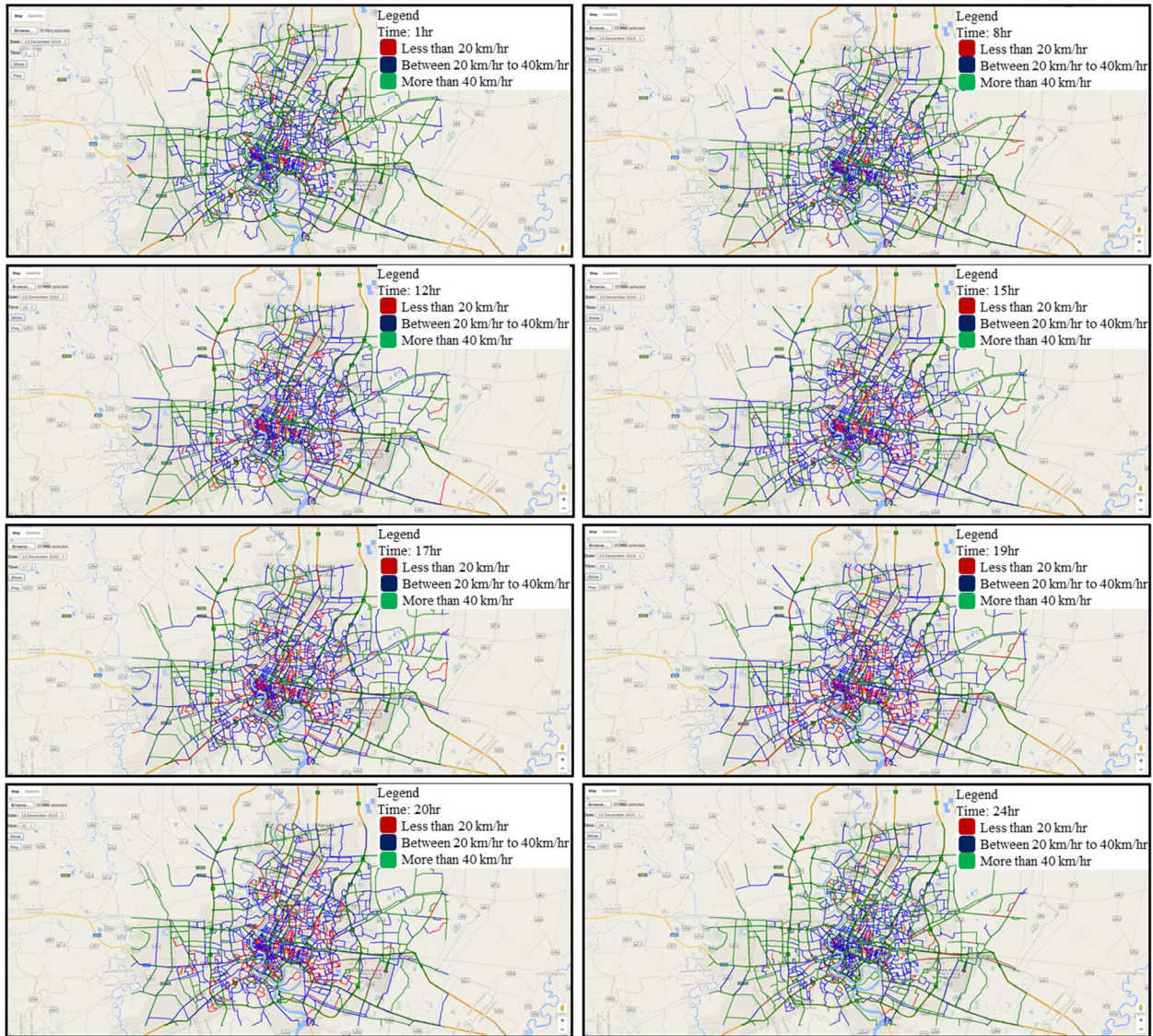
$$T_{r:\hat{p}\in t} = \begin{cases} \frac{r_{distance}}{\bar{s}_r} \\ \frac{r_{distance}}{\bar{s}_g} \dots\dots\dots \end{cases} \quad (3.7)$$

where,  $T_{r:\hat{p}\in t}$  is the road network travel time with  $r_{distance}$  as road network segment distance, such that for any point  $\hat{p}$  that appears on road network segment  $r \in R$  and grid network  $g \in G$  at time interval  $t$  during simulation, would require  $T_{r:\hat{p}\in t}$  unit time to cross or complete the road network segment.

The average speed profile at different time interval is shown in Figure 3.24. The average speed on the outer Bangkok region is higher as compared to the inner Bangkok region. This is due to the reason that inner Bangkok region has high volume of traffic as compared to sub urban region.

Managing a large volume of data requires an efficient indexing technique that would handle index, search and retrieval job (Chakka et al. 2003). Both spatial and non-spatial in indexing technique was implemented for the simulation purpose. Various spatial indexing techniques are available that based on tree data structure such as Rectangle tree R tree (Guttman 1984), similarity search tree (SS-tree) (White and Jain 1996), Sphere/Rectangle tree (SR-tree) (Katayama and Satoh 1997), Quadtree (Francis et al. 2008). However, Sort-Tile-Recursive (STR) packed R tree from Java Topological Suite (JTS) was implemented to index and search spatial data. As for non-spatial data, index and search engine named Lucene, that works on vector space model algorithm, was implemented for all query, search and retrieval task during the simulating operation (Y. Zhang and Li 2009). As for the sample data for all the data extracted, an example is for it is shown in Appendix F.





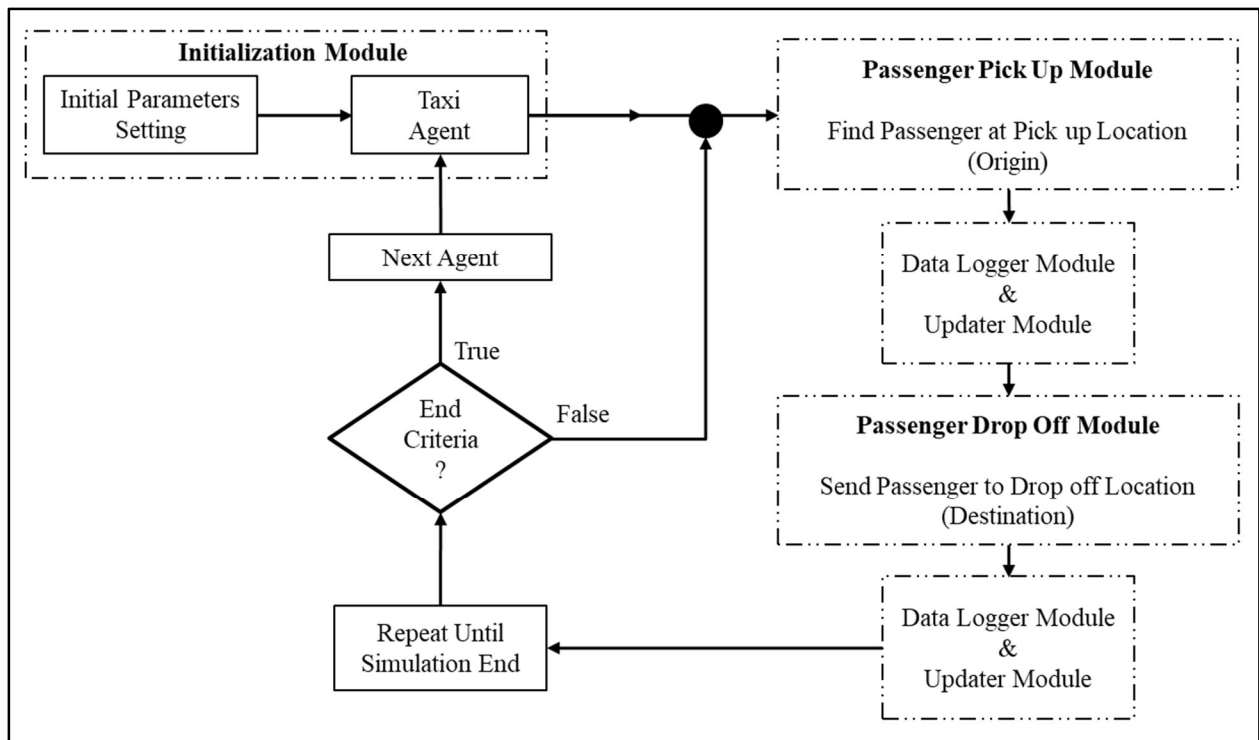
**Figure 3.24: Average speed profile on road network segment at different time interval**

The preprocessing of all the data to be utilized for simulation, including cleaning, retrieving trip information, origin-destination, stay point extraction, direction movement extraction, was conducted in Apache Hadoop/Hive large-scale distributed computing system (Witayangkurn et al. 2012, 2013). Hadoop/Hive system utilized ten nodes of which each node was a Xeon(R) CPU with 16gb of memory. The total GPS probe data preprocessed from June and July 2015 was about 2.2 billion data rows which were stored in Hadoop Distributed File System (HDFS). Each data row consisted of a GPS data points with specification as stated in Table 2.1. For spatial data processing, Apache Hive based query HiveQL (Hive Query Language) were developed including Hive UDF

(User Defined Function) and Hive UDAF (User Defined Aggregated Function). The distributed computing platform was not only for large-scale support but also for fast processing, spatial support and scalable in terms of both processing speed and storage (Witayangkurn et al. 2012).

### 3.6 Agent Based Model

An agent-based simulation model was designed to simulate the discrete event of real taxi movement as it happens in the real-world situation. The entire model was subdivided into five different submodules which were initialization module, passenger pick-up module, data logger module & updater module, passenger drop-off module, as shown in Figure 3.25.



**Figure 3.25: Agent-based simulation model**

In the agent-based simulation model, each taxi was treated as an individual taxi agent, for which they were given a specific behavior rule, based on location and time, for its entire movement. This approach makes modeling flexible, as it makes it easy to add individual variation in the behavior rule, as well as external random influences (Helbing 2012). The result is an overall characteristic

feature of the system from the collective individual entity with rule. Other features that make agent-based simulation promising is the use of modularity, where different events are separated into individual modules.

### 3.6.1 Initialization Module

In this module, various simulation spatial and non-spatial parameters were indexed. Spatial data parameters included OSM road network and grid network data, whereas non-spatial data parameters included stay point data, passenger trip data, origin-destination probability, demand probability, vacant taxi movement probability (direction probability), OSM road network, and grid network speed data. Figure 3.26 shows the initialization module of the simulation process.

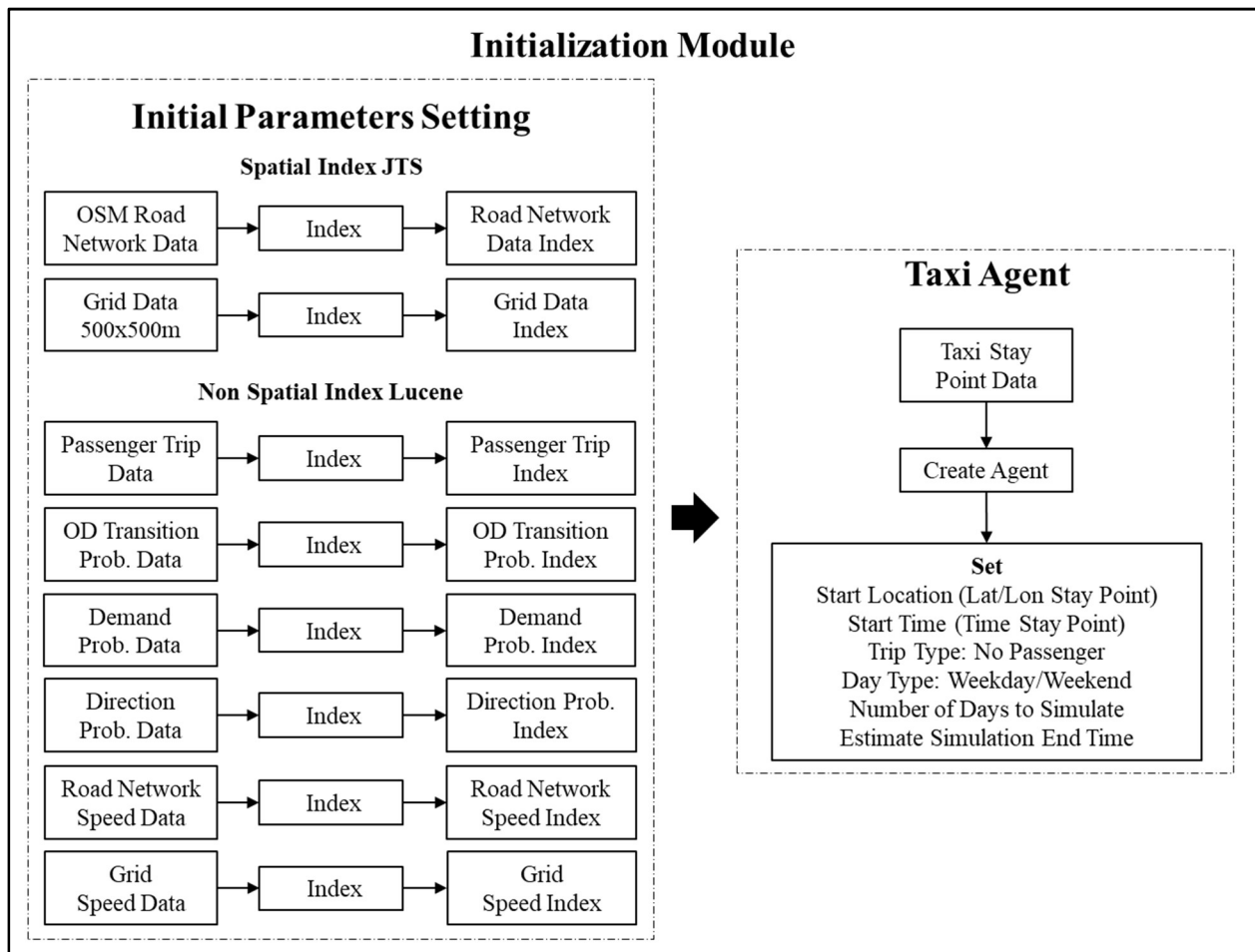


Figure 3.26: Initialization module

### 3.6.2 Taxi Agent

Each of the taxi agents was created based on the taxi stay point cluster extracted from the probe data. For each taxi agent, various parameters were set at the beginning of the simulation, including start location regarding latitude and longitude, start time, trip type, i.e., whether the taxi have a passenger or not, day type, i.e., whether simulation was for weekday or weekend, and some days or hours to run the simulation.

Based on taxi agent start time and the number of days or hours for simulation, the end or stop criteria for each taxi agent was determined. As for the starting condition, the assumption was made that the taxi agent will not have any passenger, and hence, after initialization module, the taxi agent would move onto the passenger pick-up module. Table 3.2 shows the taxi agent data which that contains Agent IMEI, Grid ID, Latitude, Longitude, Grid Geometry and Start Time as its attributes.

**Table 3.2: Agent data based on stay point cluster**

Agent IMEI	Grid Id	Latitude	Longitude	Grid Geometry	Start Time
10008773	200000042094	13.680176	100.495436	Polygon ((100.493642 13.682976, 100.498264 13.682976, 100.498264 13.678455, 100.493642 13.678455, 100.493642 13.682976))	0:27:35
10008908	200000036693	13.741863	100.605691	Polygon ((100.604570 13.746270, 100.609192 13.746270, 100.609192 13.741749, 100.604570 13.741749, 100.604570 13.746270))	0:00:01
10008917	200000031418	13.811317	100.651569	Polygon ((100.650790 13.814085, 100.655412 13.814085, 100.655412 13.809564, 100.650790 13.809564, 100.650790 13.81085))	13:23:51

### 3.6.3 Passenger Pick Up Module

In the passenger pick-up module, the taxi agent, based on location and time, would move, searching for a passenger. The searching of the passenger was made based on vacant taxi movement probability or direction probability for the grid that the taxi agent belongs to, and the time interval. Two distinct types of movement could be observed, which were staying in the same grid for searching passenger (taxi queuing event) or move adjoining grid (taxi movement event).

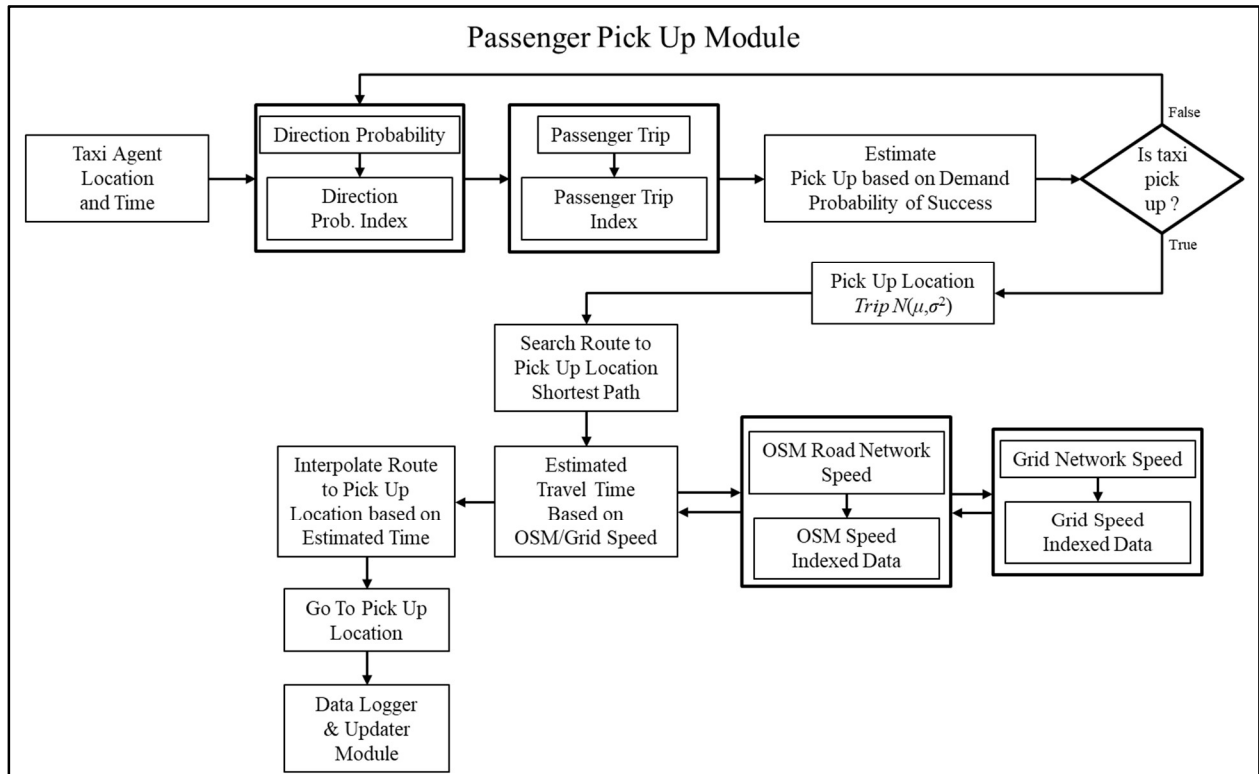
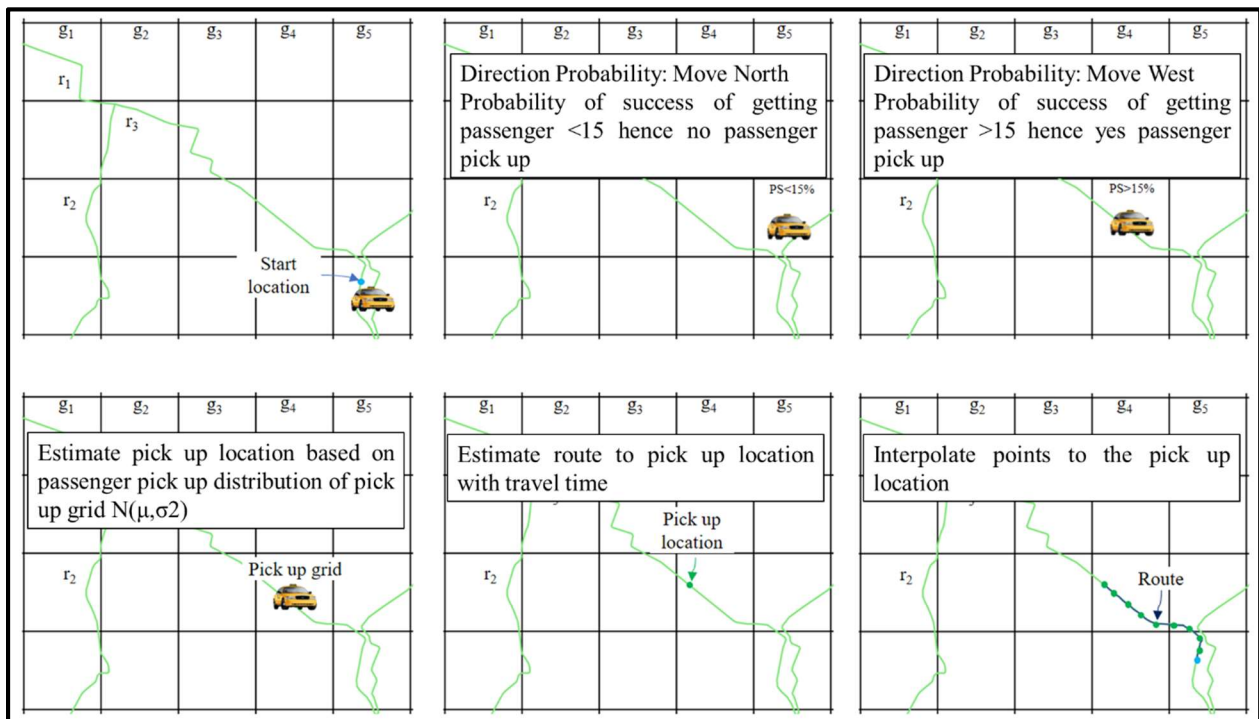


Figure 3.27: Passenger pick up module

For each taxi agent free movement at grid level, an estimation was made as to whether the taxi agent would get a passenger or not, based on demand probability of success. If there were no pick-up events, then the taxi agent would either stay in the same grid or move to an adjoining grid to look for a passenger. If there was a pick-up event in the grid, then pick-up location was estimated based on the distribution of real passenger trips that had originated in that grid. With pick-up location successfully estimated, the route to the pick-up location was implemented using the Dijkstra algorithm (Sekimoto et al. 2011) which was based on OSM network. For each road

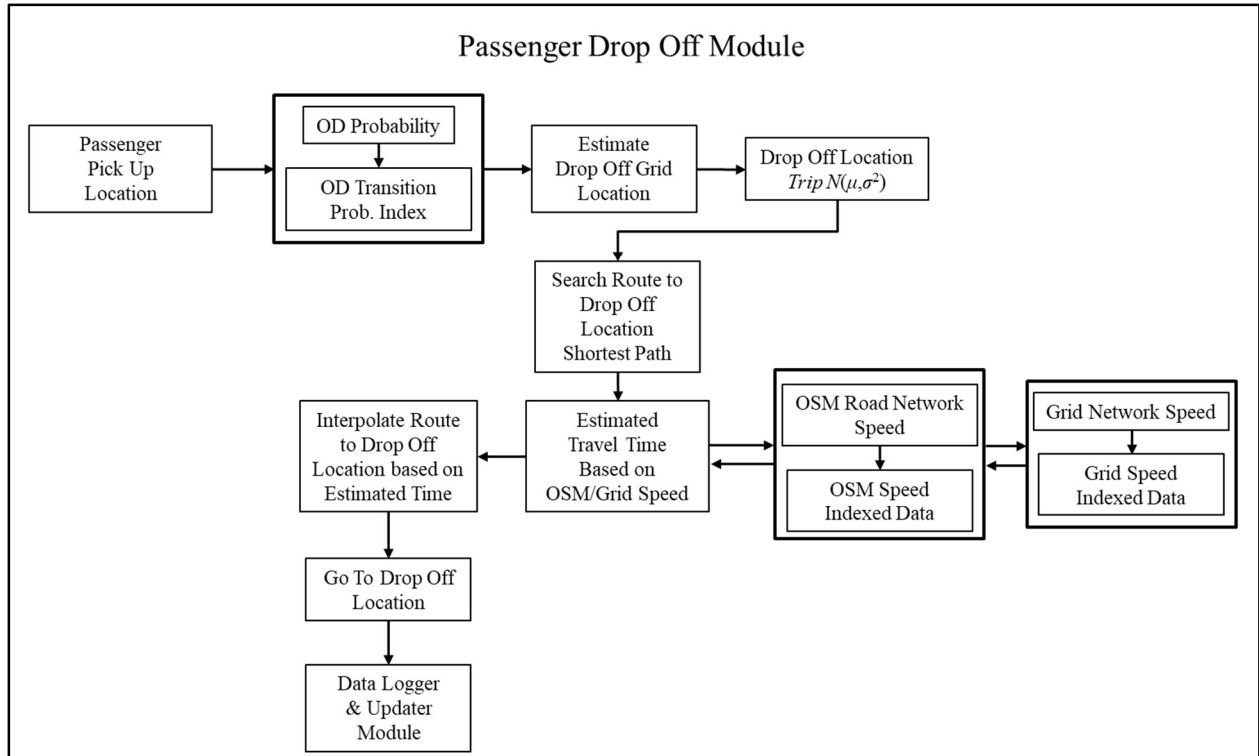
network segment, travel time was estimated on which the taxi agent would move. The cumulative travel time of each road segment of the route was then stored as passenger search time. Route interpolation, as described in (Kanasugi et al. 2013), was then implemented to generate taxi agent points  $\hat{p}$  to the pick-up location, with the sampling rate of 30 seconds. The time period was chosen as to maintain the overall trajectory (Ranjit et al. 2017) of the taxi agent, along with reducing the data storage load. Data logger module was called in to log all the interpolated points, created during the taxi agent simulation movement, on to the file system. Updater module then resets the parameters for the taxi agent, such as location, as well as time for the succeeding module. Figure 3.27 shows the detailed passenger pick-up module. As described Figure 3.28 shows the passenger pick up policy for the taxi driver when there is no passenger.



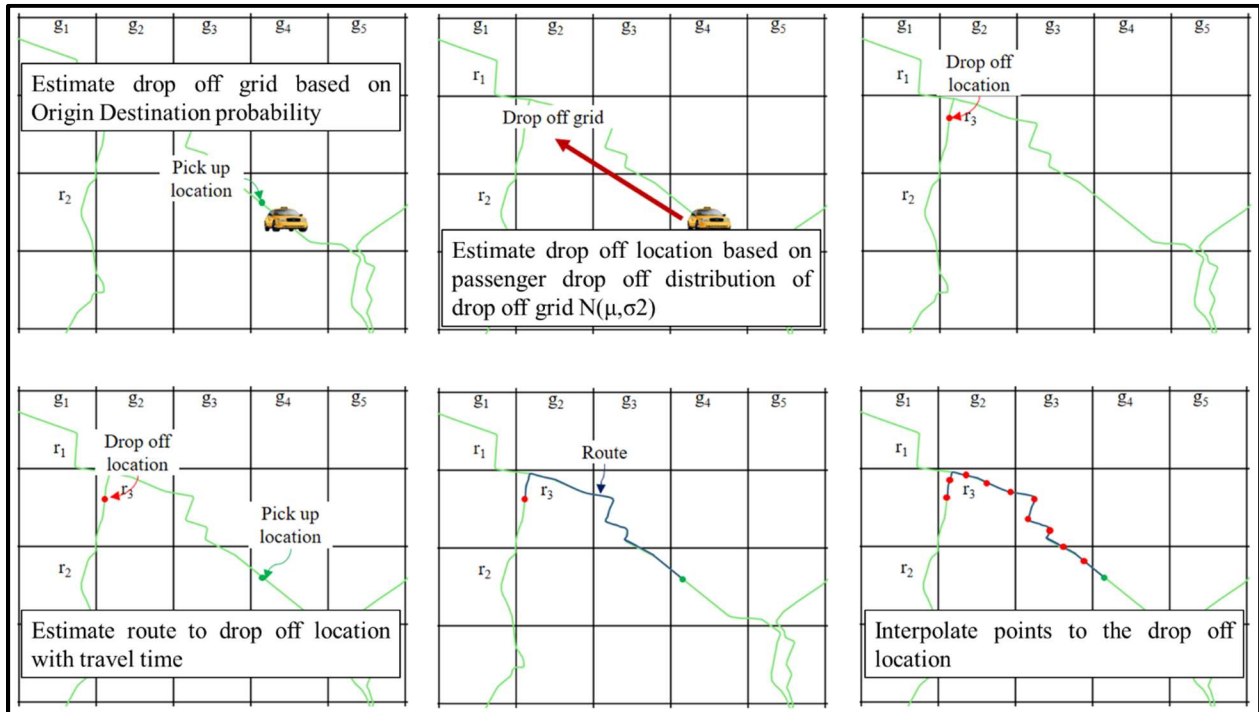
**Figure 3.28: Taxi passenger pick up policy**

### 3.6.4 Passenger Drop Off Module

Passenger drop-off module subsequently was used for estimating taxi agent drop-off location. Drop-off grid location was estimated based on trip origin-destination probability, for a given grid as well as a time interval.



**Figure 3.29: Passenger drop off module**



**Figure 3.30: Taxi passenger drop off policy**

The number of destination from a given origin could vary depending upon the location of origin. Such that in the case when the taxis pick up location is at inner city then number of destination could be large in number. In such case, distribution from top destination is selected that determined the destination for passenger drop off. With drop-off grid estimated, drop-off location was then estimated based on the distribution of real passenger trip that had destination in that grid. Following, route selection, network travel time estimation, as well as taxi agent route interpolation, was conducted as similar to the passenger pick-up module. Data logger was then called in to log all the interpolated points on to the file system along with updater module to reset parameters for the succeeding module. Figure 3.29 shows the detailed passenger drop-off module. As described Figure 3.30 shows the passenger drop off up policy for the taxi driver with passenger

### 3.6.5 Data Logger and Updater Module

The purpose of the data logger module was to log all the simulated data generated during both passenger pick-up module and passenger drop-off module. Following the data logger module, updater module was called in, as shown in Figure 3.31.

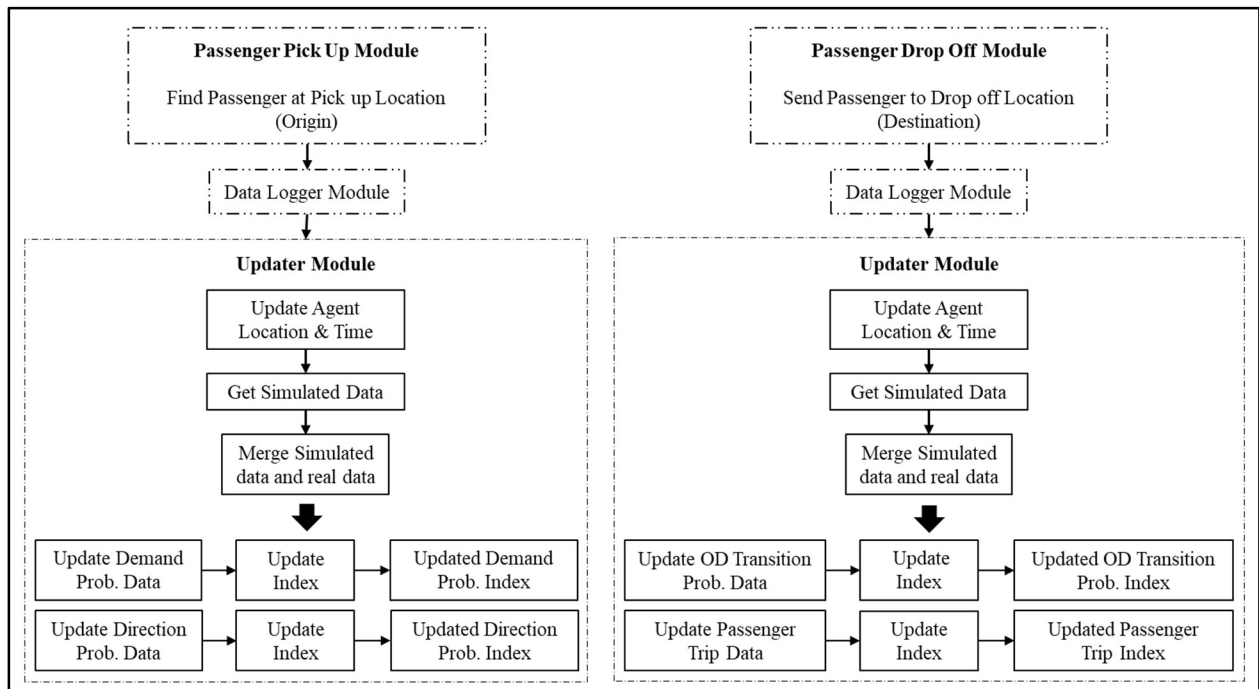


Figure 3.31: Data logger and updater module



The updater module served mainly two purposes. The first purpose was to update the individual local parameter of an agent, such as agent location and time. The second purpose was to update the global parameters, which included taxi demand probability data which is regarded as the improvement in model, direction probability data, taxi origin destination probability data, and passenger trip data. The global parameter could be updated by merging the simulated data with the real dataset. The new taxi demand probability data, direction probability data, taxi origin destination probability data, and passenger trip data, then could be computed, based on merged simulated and real dataset. The index of the parameters then needs to be updated, which would be used by the agents subsequently during simulation. The update on global parameter provided a mechanism where agent could interact with each other indirectly.

### **3.7 Model Evaluation**

Model evaluation is one of the important aspect when designing and implementing the model to any social use. Without proper evaluation of the model the accuracy of the model cannot be judges. Simulation model is a process that shows real world system which shows behavior or help evaluate different strategies (Gómez-Sanz et al. 2010). Agent-based modeling as suited for modeling individual centered dynamics poses many parameter that can influence the behavior of the model (Reuillon et al. 2015). Different evaluation method is available to evaluate the model one of which is the use of statistical analysis for the evaluation proposes (Kleijnen 1996). Other approach of evaluating the model includes the holistic approach as described in (Bharathy and Silverman 2012). The simulation was conducted in two scenarios, i.e., weekday and weekend, each for an entire day, such that overall properties of simulated taxi service behavior within the city was kept intact, as that with the real taxi service, by adjusting various parameters within the simulation process. Various property comparisons, based on distribution, were conducted between the simulated taxi agent data and the real taxi data, that showed the level of similarity between them. The overlapping coefficient, which is defined as a measure of the agreement between two probability distributions (Inman and Bradley 1989), was computed to identify the similarity measurement with a scale of 0 to 1. The measured value of 1 indicated a perfect match, while the measured value of 0 indicated no interaction between the distribution. Four different taxi attributes that included trip generated, trip generated per grid, trip time, and trip distance were compared together regarding their

distribution, whereas average taxi speed was compared for hourly variation. Finally, occupancy between the simulated and real taxi data was also evaluated.

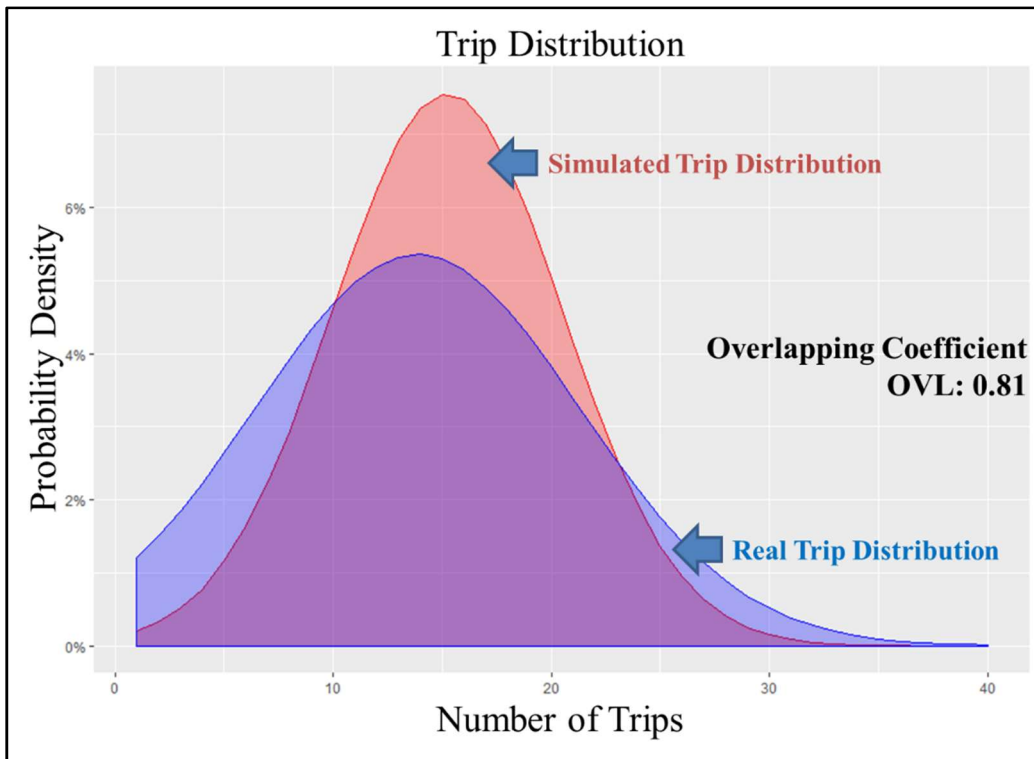
### 3.7.1 Distribution Overlapping Coefficient

The weekday’s distribution comparison is shown in Figure 3.32-3.35. Trip distribution comparison between simulated taxi data and real taxi data showed the overlapping coefficient of 0.81, which indicated some trips generated from the simulation were a close match with the real data. Similarly, a number of trips generated per grid distribution showed a very high overlapping coefficient of 0.98, indicating high similarity for the trip generated on the grid level.

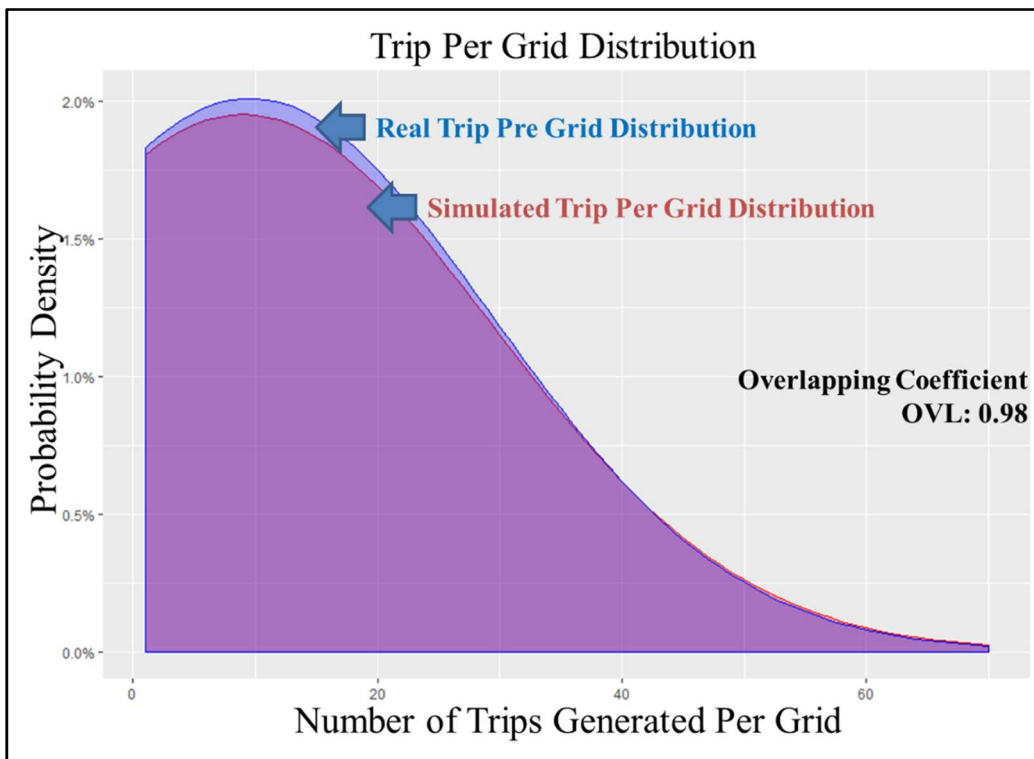
As for the trip time distribution, regarding minutes, an overlapping similarity of 0.93 was obtained, which showed significant similarity between simulated trip time and real trip time. Finally, for trip distance, regarding kilometers, a distribution of overlapping similarity of 0.88 was obtained between simulated trip time and real trip time. The significant overlapping similarity for the weekday simulation suggested that the simulated taxi agent emulated the real taxi, keeping overall taxi behavior intact. Table 3.3 shows distribution properties of the four compared attributes of taxi behavior for the weekday simulation.

**Table 3.3: Weekday simulated trip data vs real trip data comparison**

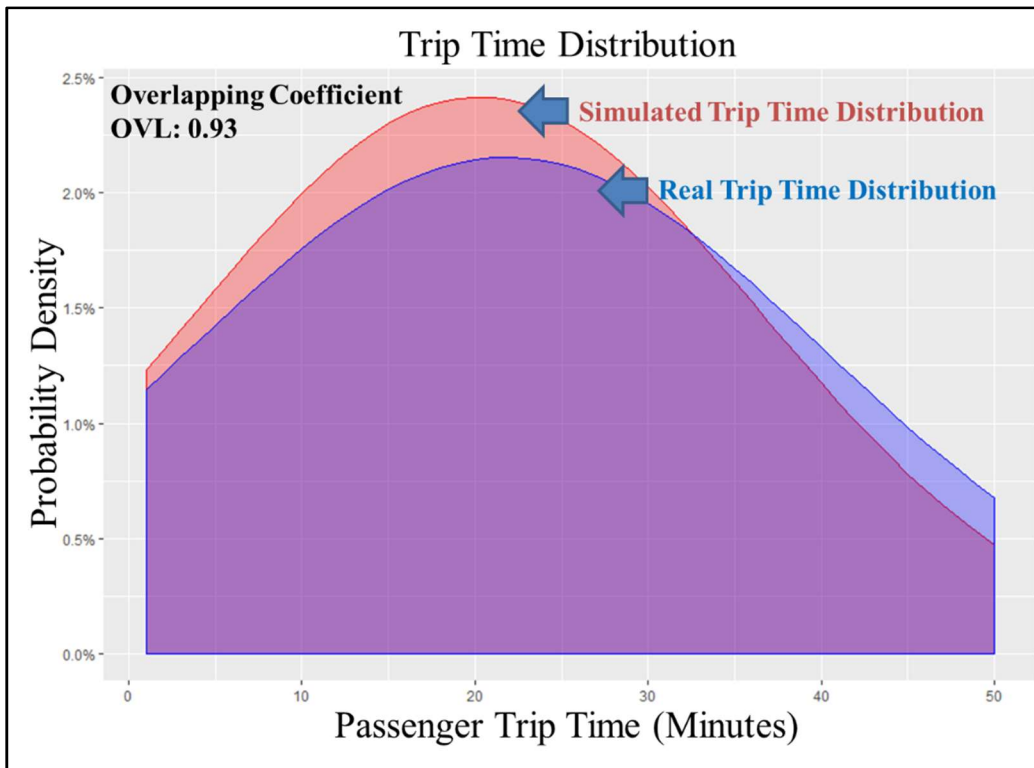
Weekday					
Type	Simulated Data		Real Data		Overlapping Coefficient
	Mean	Standard Deviation	Mean	Standard Deviation	
Trip Count	15.24	5.27	13.87	7.44	0.81
Grid Trip Generated	9.02	20.42	9.57	19.82	0.98
Passenger Trip Time (min)	20.18	16.50	21.81	18.50	0.93
Passenger Trip Distance (km)	9.09	7.87	10.31	9.78	0.88



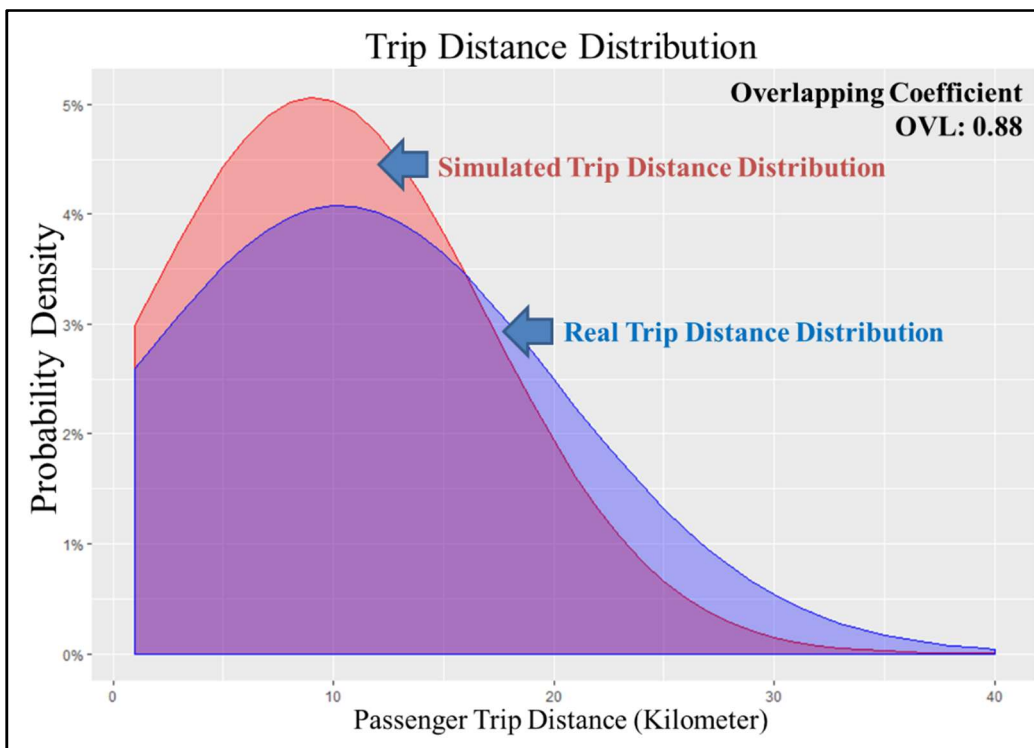
**Figure 3.32: Weekdays trip distribution comparison**



**Figure 3.33: Weekdays trip per grid distribution comparison**



**Figure 3.34: Weekdays trip time distribution comparison**

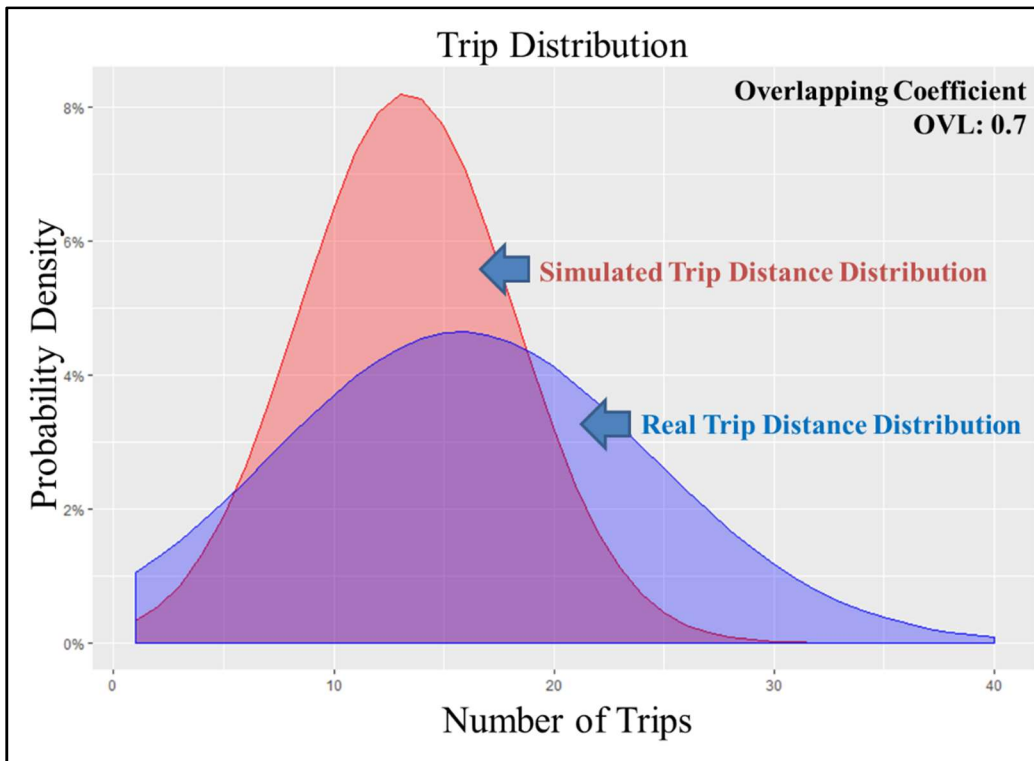


**Figure 3.35: Weekdays trip distance distribution**

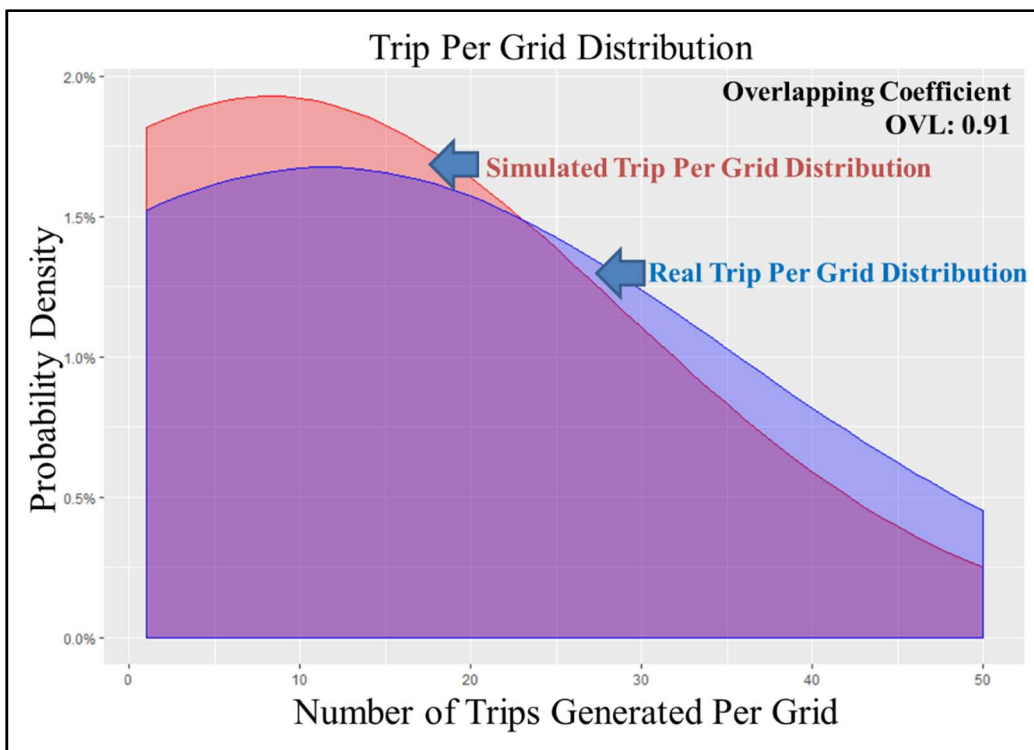
The weekend distribution comparison is shown in Figure 3.36-3.39. Trip distribution comparison between simulated taxi data and real taxi data showed the overlapping coefficient of 0.7, which indicated a number of trips generated from the simulation were in accord with the real data. Similarly, a number of trips generated per grid distribution showed a high overlapping coefficient of 0.91, indicating high similarity for trips generated on the grid level. For trip time distribution, with respect to minutes, the overlapping coefficient of 0.96 was obtained, which showed a significant similarity between simulated trip time and real trip time. Finally, for trip distance distribution, regarding kilometers, an overlapping similarity of 0.86 was obtained between simulated trip time and real trip time. The significant overlapping similarity for the weekend simulation also suggested that the simulated taxi agent emulated the real taxi keeping overall taxi behavior intact. Table 3.4 shows distribution properties of the four compared attributes of taxi behavior for the weekend simulation. As described previously, only those trips that had origin and destination in Bangkok and surrounding provinces were considered to construct the OD matrix, and subsequently, OD probability. Hence, for both weekday and weekend, in the overlapping coefficient similarity comparison, only those trips within Bangkok and surrounding provinces were considered. Out of all the real passenger trips within Bangkok and surrounding provinces, 97% of the trip had a trip time of less than 2 hour, and 98% of the trip had a trip distance less than 100 km. This implies that the simulated result obtained could emulate taxi behaviors for a trip distance within 100 km, and trip time within 2 hours.

**Table 3.4: Weekend simulation trip data vs real trip data comparison**

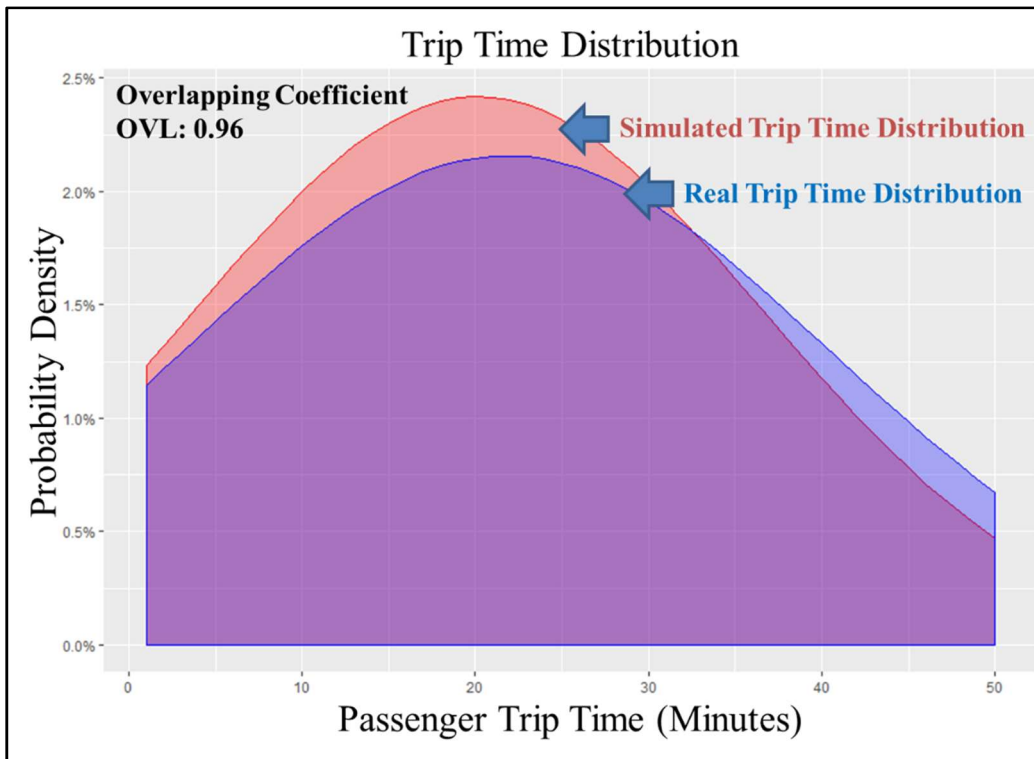
Weekend					
Type	Simulated Data		Real Data		Overlapping Coefficient
	Mean	Standard Deviation	Mean	Standard Deviation	
Trip Count	13.31	4.86	15.78	8.58	0.70
Grid Trip Generated	8.18	20.68	11.50	23.81	0.91
Passenger Trip Time (min)	19.17	16.13	20.37	16.70	0.96
Passenger Trip Distance (km)	9.01	7.91	10.74	10.02	0.86



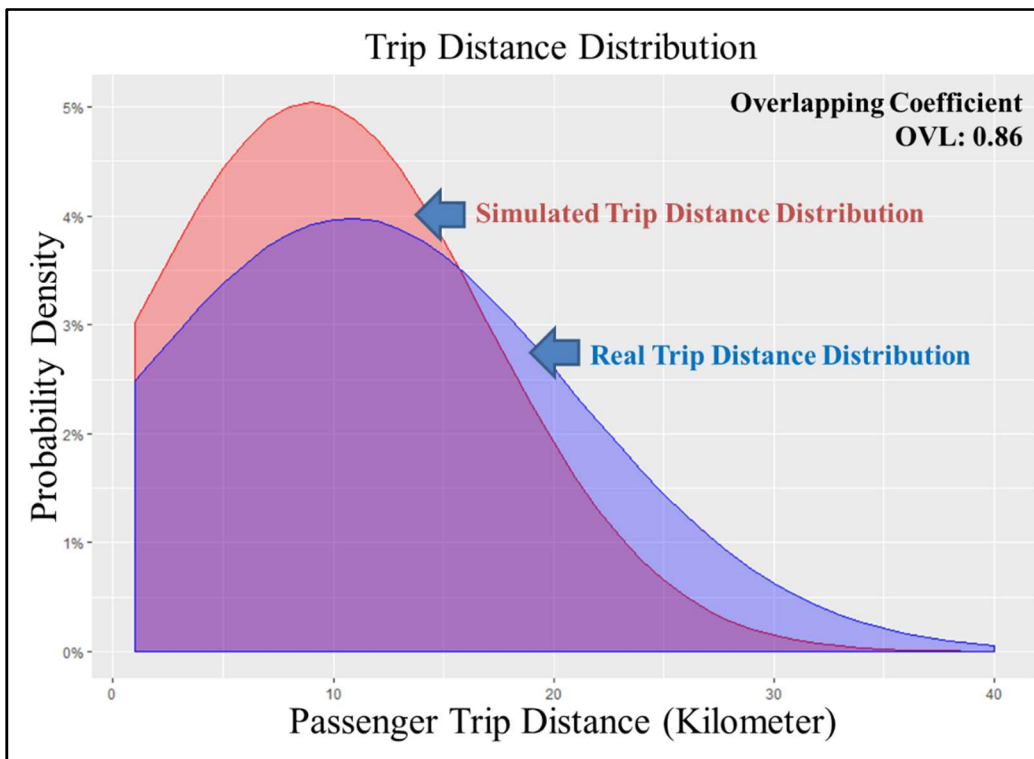
**Figure 3.36: Weekends trip distribution comparison**



**Figure 3.37: Weekends trip per grid distribution**



**Figure 3.38: Weekends trip time distribution**

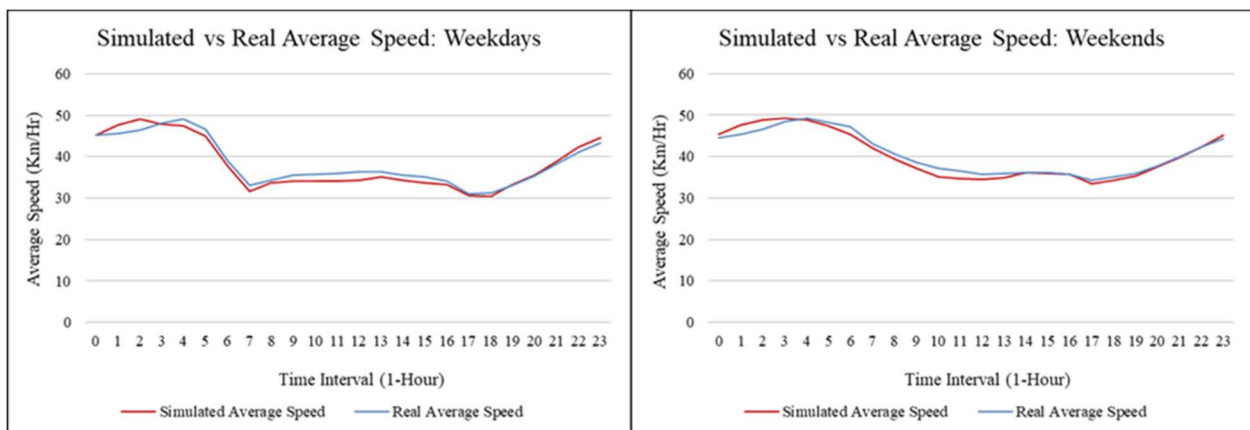


**Figure 3.39: Weekends trip distance distribution**

The simulated data results in parameters i.e., trip count, grid trip generated, passenger trip time (min), and passenger trip distance (km), as compared to the real data result parameters, was marginally lower, as shown in Table 3.4. However, simulated data result parameters could be maintained by adjusting the demand probability success for which the current threshold value was set empirically to 15%. In addition, the higher value of standard deviation for both simulated weekday and weekend results especially for the grip trip generated and passenger trip time was obtained as the distribution was computed at the grid level for which grip trip generated within the inner city would have obtained more trip as compared to the grid which was located at the outskirts of the city. Similarly, passenger trip time would also vary depending upon the individual trip generated, which resulted in the high standard deviation in simulated result.

### 3.7.2 Average Speed Hourly Variation

OSM road network based average speed comparison was conducted for time intervals of 1 hour between simulated and real datasets for both weekday and weekend. Figure 3.40 shows the hourly variation of average speed for the entire region of Bangkok and surrounding provinces. Comparison of an average speed showed an  $R$  squared value of 0.96 for the weekday comparison and  $R$  squared value of 0.97 for the weekend's comparison. The high  $R$  squared value for both weekday and weekend simulated data suggested simulated taxi agents could keep the real taxi properties intact. As mentioned previously, all query, search, and retrieval tasks were conducted over grid network.



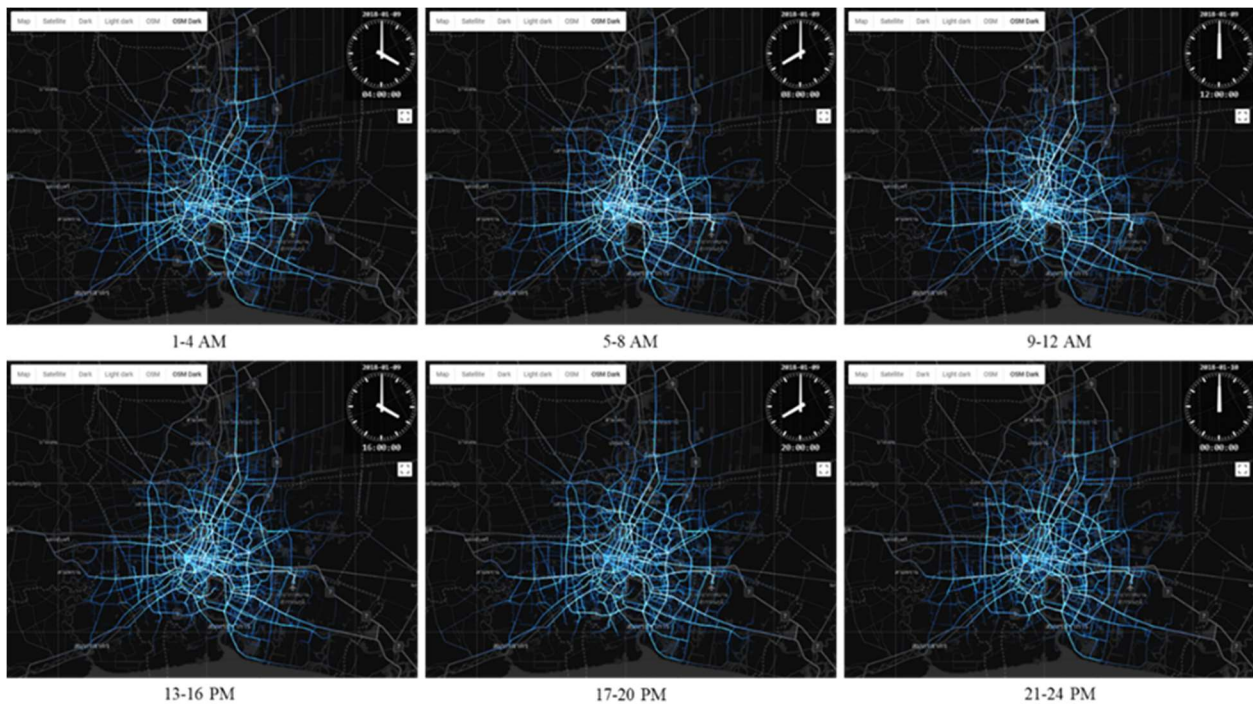
**Figure 3.40: Average speed variation concerning hourly time interval**



The routing was conducted with the shortest path algorithm and route interpolation conducted, based on trip time interval over the OSM road network. The simulated data is shown in the Table 3.5. The simulated data is recorded during simulation for seven attributes where are IMEI, Latitude, Longitude, Speed, Meter, Timestamp and Day Type. Current simulation is done in the standalone server however using the distributed computing platform could enhance the performance in terms of simulation time. The sample of the simulated trajectory for the weekday data at a time interval of every 4 hour is shown in Figure 3.41.

**Table 3.5: Sample simulated taxi data**

IMEI	Latitude	Longitude	Speed	Meter	Timestamp		Day Type
10016792	13.874258	100.67198	11.88	0	2015-06-01	4:50:04	Weekdays
10016792	13.873953	100.672217	11.88	0	2015-06-01	4:50:34	Weekdays
10016792	13.873953	100.672217	34.79	1	2015-06-01	4:50:34	Weekdays
10016792	13.873676	100.669998	34.79	1	2015-06-01	4:51:04	Weekdays
10016792	13.87193	100.668309	34.79	1	2015-06-01	4:51:34	Weekdays

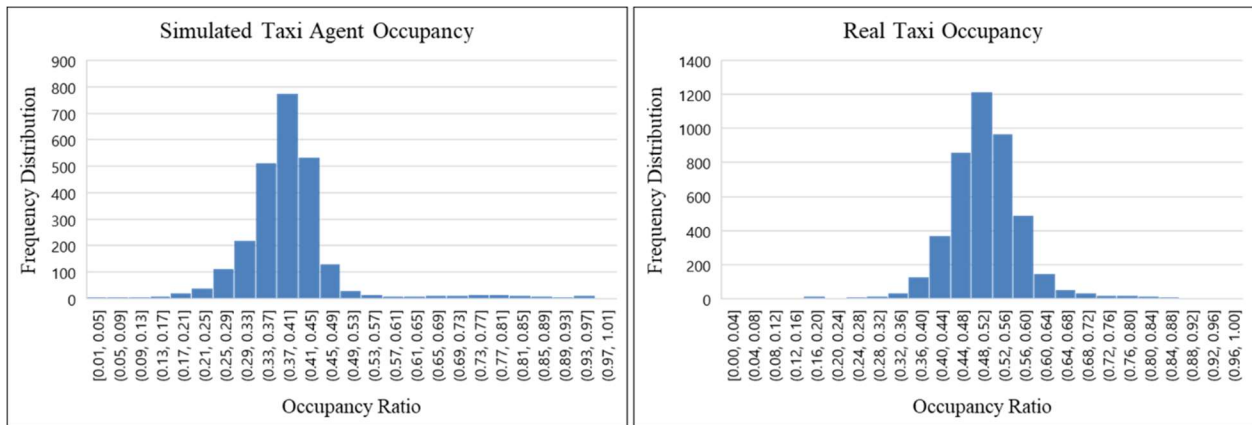


**Figure 3.41: Simulated taxi agent trajectory visualization for weekdays simulation**

### 3.7.3 Taxi Occupancy Evaluation

One of the properties that characterizes taxi service behavior is its occupancy, which is defined as the ratio of taxi driving time with a passenger to total driving time (Lv et al. 2017a). As the probe data captures the taxi movement constantly which passenger status occupancy rate could be extracted from the probe data (Leduc 2008; Z. Zheng et al. 2014). Occupancy ratio also shows the driver satisfaction as the ratio number of minutes the taxi is occupied in a given location to total minutes the taxi is available (Balan et al. 2008). Figure 3.42 shows the frequency distribution of taxi occupancy ratio for the simulated taxi agent and real taxi data.

The occupancy ratio analysis showed for simulated taxi agent data occupancy was clustered around 37–41%, whereas for the real taxi data, occupancy ratio was clustered around 48–52%. Though simulated taxi data showed slight underestimation regarding occupancy ratio; the overall distribution was kept similar to the real taxi data.



**Figure 3.42: Taxi occupancy. Left: Simulated taxi agent data; Right Real taxi data**

### 3.7.4 Taxi Fare Evaluation

The taxi fare structure is the key component that affectively determines the income for the taxi driver. In Bangkok, Thailand taxi fare structure has gone through restructuration from the year

2015. The new fare is prepared to make taxi driver at benefit as compared to the fare structure pre-2015. The old and the new fare structure is shown in Table 3.6.

**Table 3.6: Taxi fare structure in Bangkok, Thailand**

<b>Taxi Fare Structure (Old)</b>	
<b>Distance in Kilometer</b>	<b>Fare (Thai Baht)</b>
0 – 2	35 THB
2 – 12	5.0 THB
12 – 20	5.5 THB
20 – 40	6.0 THB
40 – 60	6.5 THB
60 – 80	7.5 THB
80+	8.5 THB
Minute (slow traffic)	1.5 THB
<b>Taxi Fare Structure (New from 2015)</b>	
<b>Distance in Kilometer</b>	<b>Fare (Thai Baht)</b>
0 – 1	35 THB
1 – 10	5.5 THB
10 – 20	6.5 THB
20 – 40	7.5 THB
40 – 60	8.0 THB
60 – 80	9.0 THB
80+	10.5 THB
Minute (slow traffic)	2.0 THB

# Taxi Income Distribution Weekday & Weekend

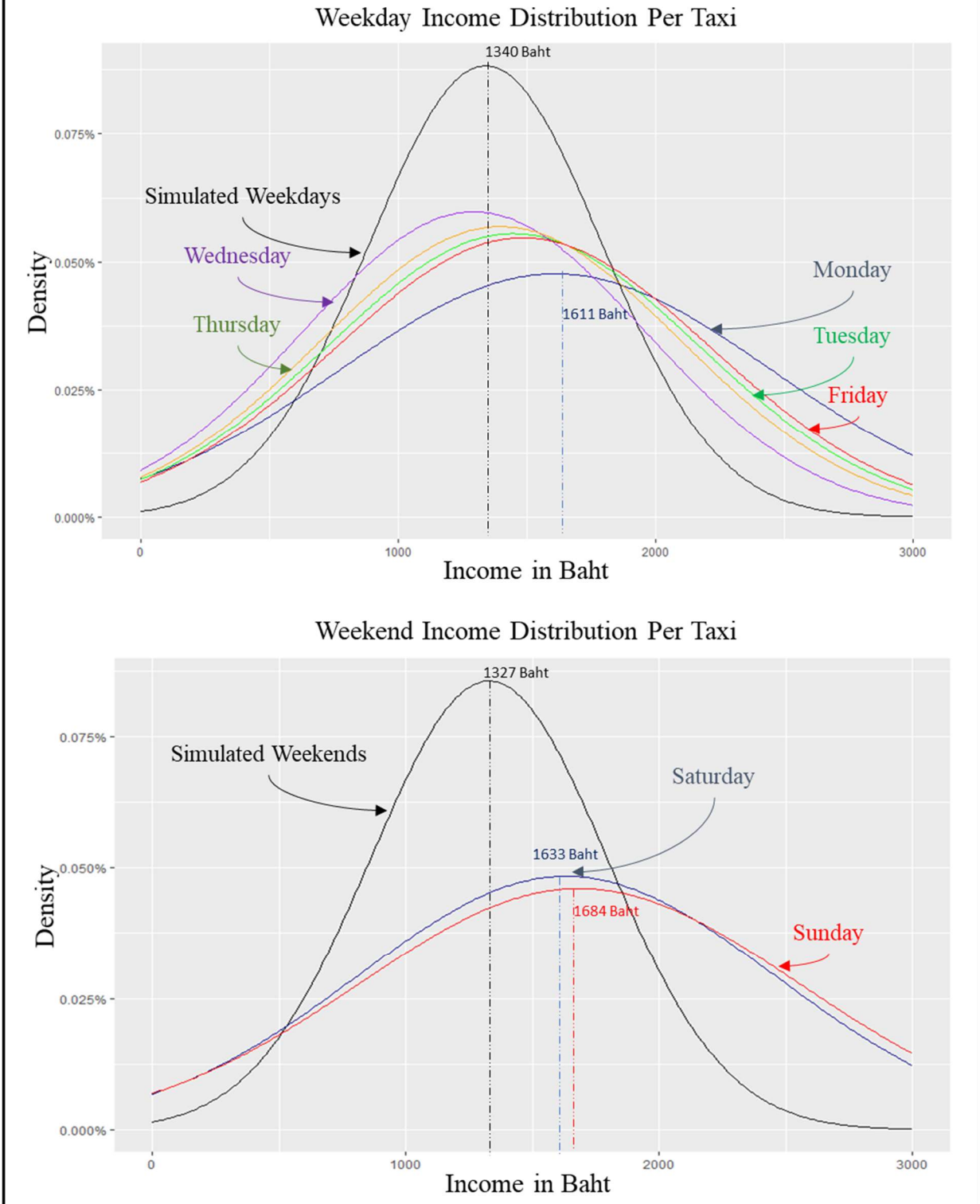


Figure 3.43: Taxi income distribution for weekday and weekend

As the data obtained is June and July 2015, new fare structure was used to evaluate the taxi driver income. Figure 3.43 shows the taxi driver income distribution for both weekdays and weekends. However, taxi driver income was under estimated as shown in Table 3.7, in both weekdays and weekend simulation as compared to real data which could be the result of using shortest path algorithm during simulation.

**Table 3.7: Driver income distribution**

<b>Day</b>	<b>Average Income in Baht</b>
Weekdays Simulated	1340
Monday	1611
Tuesday	1447
Wednesday	1294
Thursday	1398
Friday	1486
<b>Day</b>	<b>Average Income in Baht</b>
Weekend Simulated	1327
Saturday	1633
Sunday	1684

### 3.8 Model Improvement

The agent-based simulation model for taxi behavior can be subject for the improvement based on multiple criteria at different stages of the simulation. The evaluation of the taxi behavior simulation with multiple indicator i.e. trip distribution, trip per grid distribution, trip time distribution, trip distance distribution showed significant level of similarity between the simulated taxi probe data and the real probe data. However, some indicator did poses some level of variance between the simulated probe data and the real probe data despite having reasonable overlapping coefficient. To overcome the issue improvements were added to the existing simulation model. In addition, simulation itself is heavy in terms of computation time. Varying variable within the simulation to

re-simulate multiple time could be time consuming. Moreover, varying and increasing the number of agents would further add computation overhead to the model. To address the issue, a distributed computation platform was utilized that worked on number of machine working together to complete the simulation. The improvement not only help improve the computation performance but also number of agent could be easily changed for the simulation model itself.

### 3.8.1 Demand Update

In the Agent-based simulation model the probability of success which shows how likely the taxi could get the passenger was constant with probability which was set at the start of the simulation. The problem with this setting is regardless of the number of passenger, for a given grid and time interval all the taxi could or could not get the passenger based on the probability of success value. If probability of success was greater than or equal to 15% threshold value, the all the taxi would get the passenger and if less than 15 % threshold value the taxi would not get the passenger.

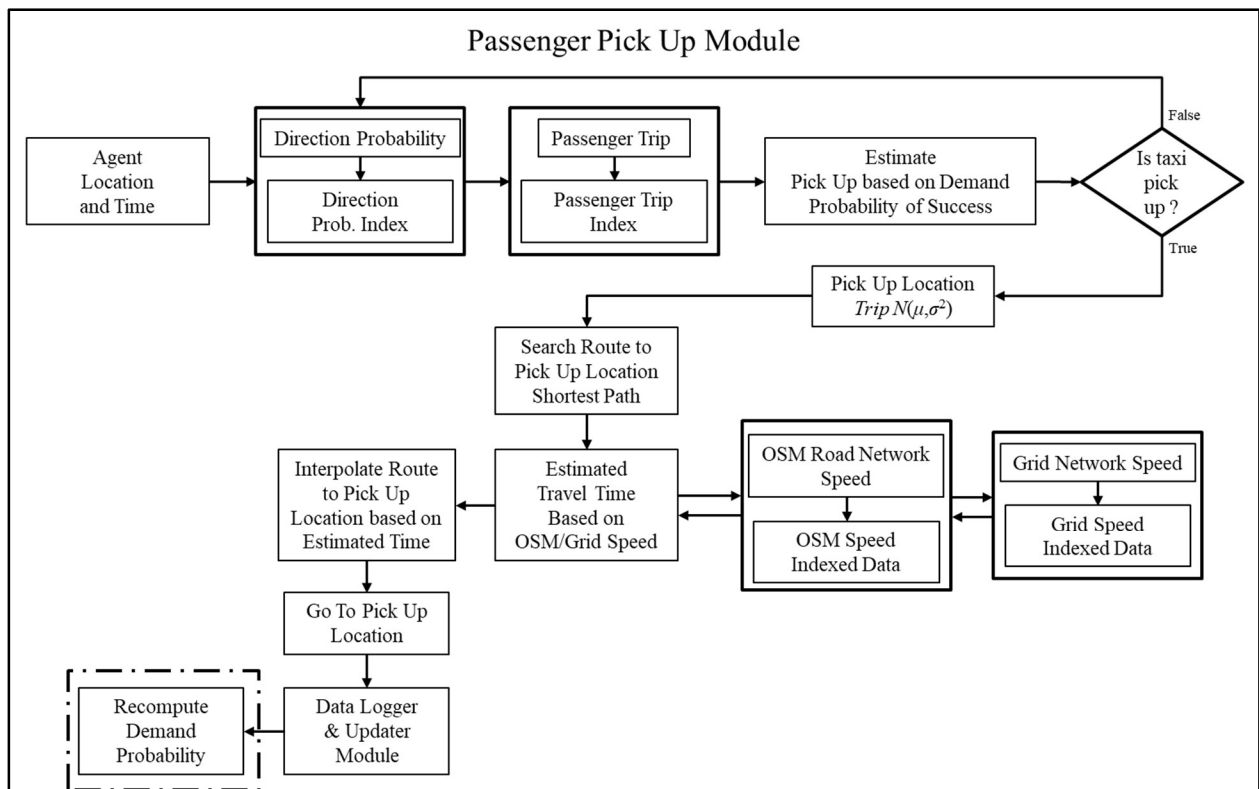


Figure 3.44: Passenger updated pick up module

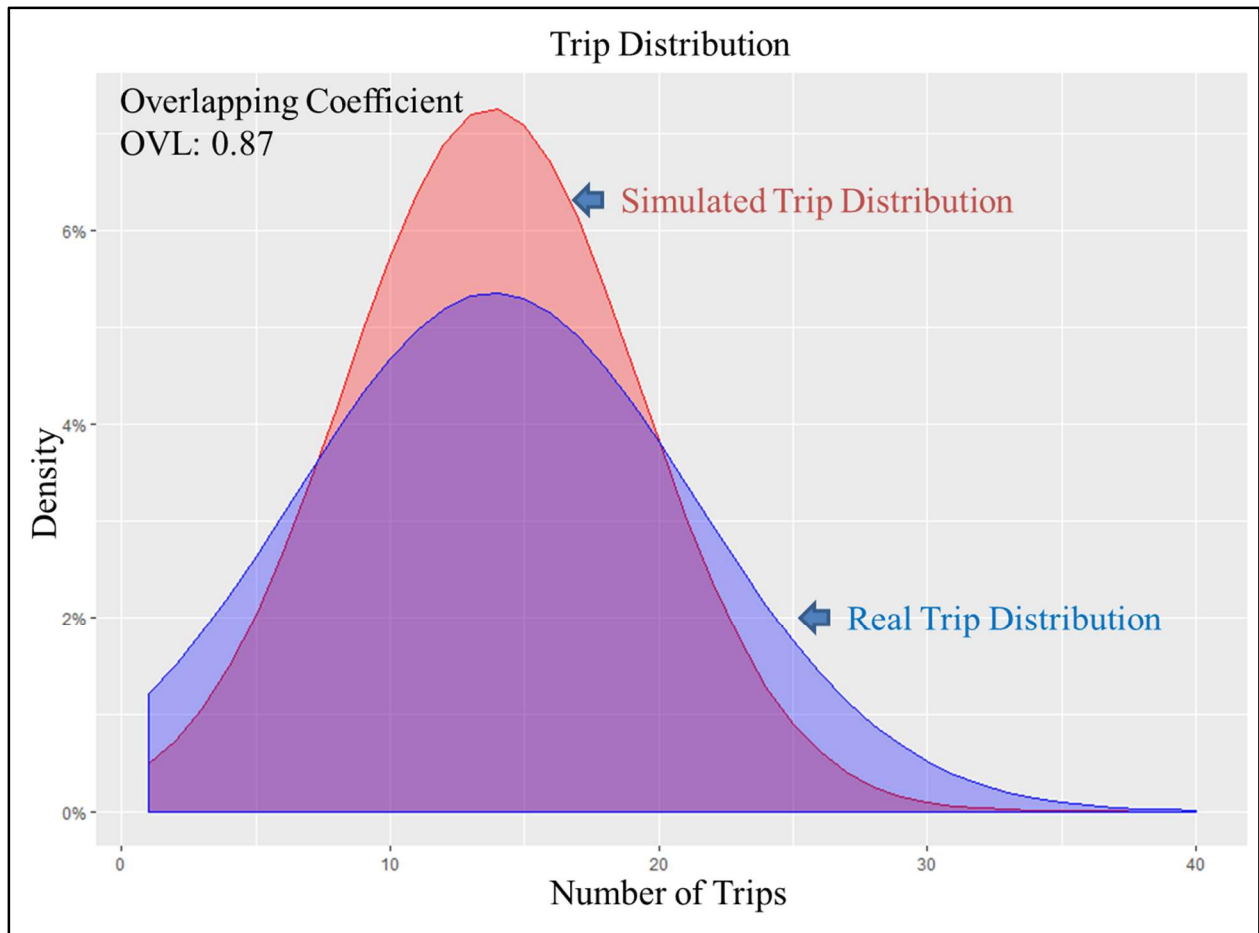
However, in real practice the demand itself could change within the given grid and time interval. Hence the dynamic demand probability of success was introduced. The idea for the dynamic demand probability of success was to re-evaluate the probability of success as taxi start getting passenger in a given grid and time interval. In this regard, the demand probability was recomputed during the runtime simulation such that for the given grid and time interval, as the taxi starts to get the passenger, probability of success was recomputed and reduced as shown in the equation 3.8.

$$\forall g \in G_t, Pr(dm)_g = \frac{O_g - 1}{V_g - 1} \dots \dots \dots 3.8$$

where  $Pr(dm)_g$  is the probability of success for all grid  $g \in G$  at time interval  $t$ , such that  $O_g$  and  $V_g$  are the total number of demands generated and total number of recorded vacant taxis at grid  $g \in G$  and time interval  $t$ , respectively.

The result was that rather than all the taxi getting the passenger with respect to initial probability of success, the recomputed probability of success makes that not all the taxi gets the passenger even if they are in the same grid and time interval which is more natural way of taxi behavior simulation. As this the passenger pick up module was also updated to accommodate the re-computation of probability of success. The updated passenger pick-up module is shown in Figure 3.44. The setting would provide more natural behavior that allows only certain number of passenger to be picked up by the taxi agent in the given grid and time interval.

The comparison results between the real taxi data and the simulated taxi agent data with updated pick up module showed increase in the distribution overlapping coefficient as compared to pick up module with constant probability of success. Figure 3.45 shows the trip distribution for the weekdays simulated data. As evaluated the mean and deviation for simulated data was measured at 13.78 and 5.48 while for the real data mean and deviation was measured as 13.87 and 7.44. The introduction of the dynamic probability of success helped make mean to very close proximity. However, there was still some difference between the deviation.

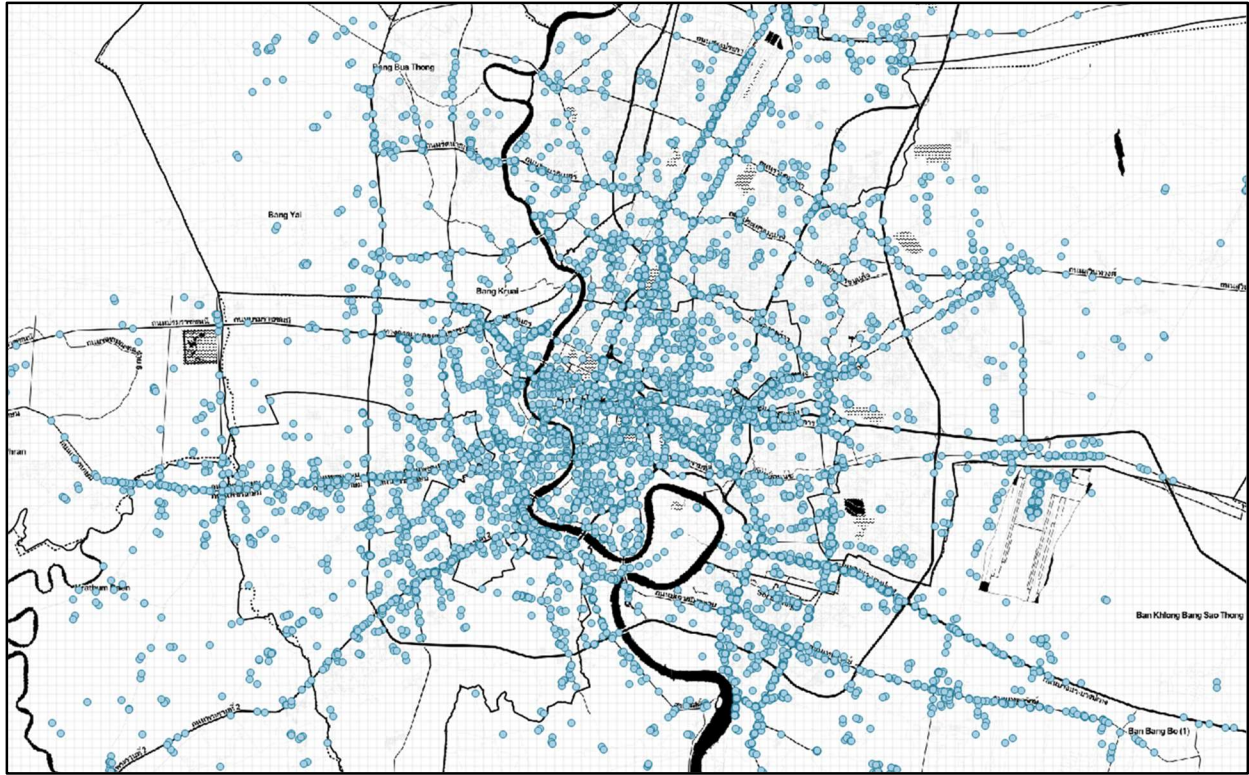


**Figure 3.45: Trip distribution for the updated pick up module (day type = weekdays)**

### 3.8.2 Distributed System Implementation

The final improvement introduced to the simulation model is the implementation of the overall simulation process in the distributed system. A 10 node hadoop/hive distributed cluster was utilized for the overall computation. Hadoop/Hive user defined function was built to carry out the operation for the simulation. As of this, simulation was tested for different number of taxi agent which were derived from the probe taxi stay point cluster. Taxi agent of 3000, 5000 and 10000 were simulated in the distributed system and evaluated accordingly. Figure 3.46 shows the taxi agents that are distributed along the Bangkok and the surrounding provinces.





**Figure 3.46: Taxi agent distributed across Bangkok and surrounding provinces**

## CHAPTER 4

### OPTIMIZATION OF TAXI OPERATION

#### 4.

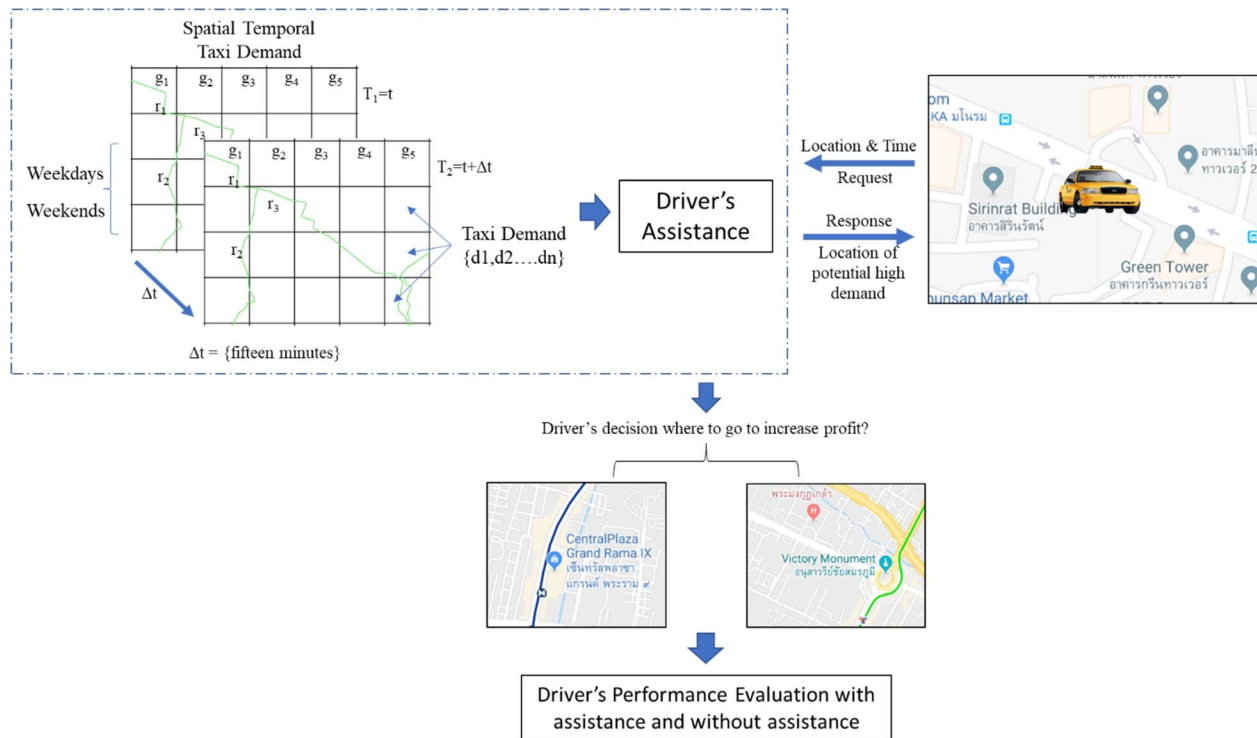
##### 4.1 Introduction

Optimization is the process that finds the best possible way to use the available resources while not violating any given constraints (Lindfield and Penny 2017). Optimization are key to solve various mathematical problem in many disciplines (Rothlauf 2011) that would help find solutions which are optimal with regards to the goal. In this regards taxi operation optimization is also a key aspect on how taxi drivers could improve or increase their income.

As described in the Chapter 1, there are existing problems associated with the taxi operation in Bangkok region. The problems are associated with the taxi drivers for not getting enough passenger and hence low income. The vacant taxis are the waste of fuel and money as well as problem to the environment (Lv et al. 2017b). On the other hand, there are problem within the passenger for being refusal of the taxi service. The simulation model as described in Chapter 3 could help male better understand of the existing taxi operation. While the simulation model does the help make improvement of the service within itself the job is left for the optimization model. The optimization of the taxi operation could help reduce or minimize the issue faced by both passenger as well as driver. Hence, the advantage of optimization of the taxi operation is not just beneficial for the taxi driver but also for the passenger as well.

Different method for the optimization of the taxi operation have been proposed in the past literature and researches. In recent years the utilization of the GPS trace for making prediction on a mobility pattern (Castro et al. 2012) that included mobility of the taxi. Analyses of spatiotemporal behavior of taxi services could make a better use for urban planners and managers (Deng and Ji 2011). A time location sociality model where three dimensional properties of the city dynamic is considered to predict the distribution of the passenger is proposed by (Yang et al. 2016). The model recommended top N locations to the driver based on the historical traffic data where drivers would have high chance of finding the passengers. A two layer model, in which the first layer model

provide decision making recommendation to the taxi driver which zone to go for passenger pick up and the second layer model provided routing decision for the taxi driver is proposed by (Tang et al. 2016). The model used DBSCAN algorithm to cluster the of pick up and drop off record and described the attractiveness of pick up location with Huff model. As for the routing behavior the model implemented Path Size Logit (PSL) model considering travel time and distance as well as path size with delay in intersections. On the other hand, demand prediction for the next time frame has been proposed using deep learning algorithm as quoted in (Yao et al. 2018).

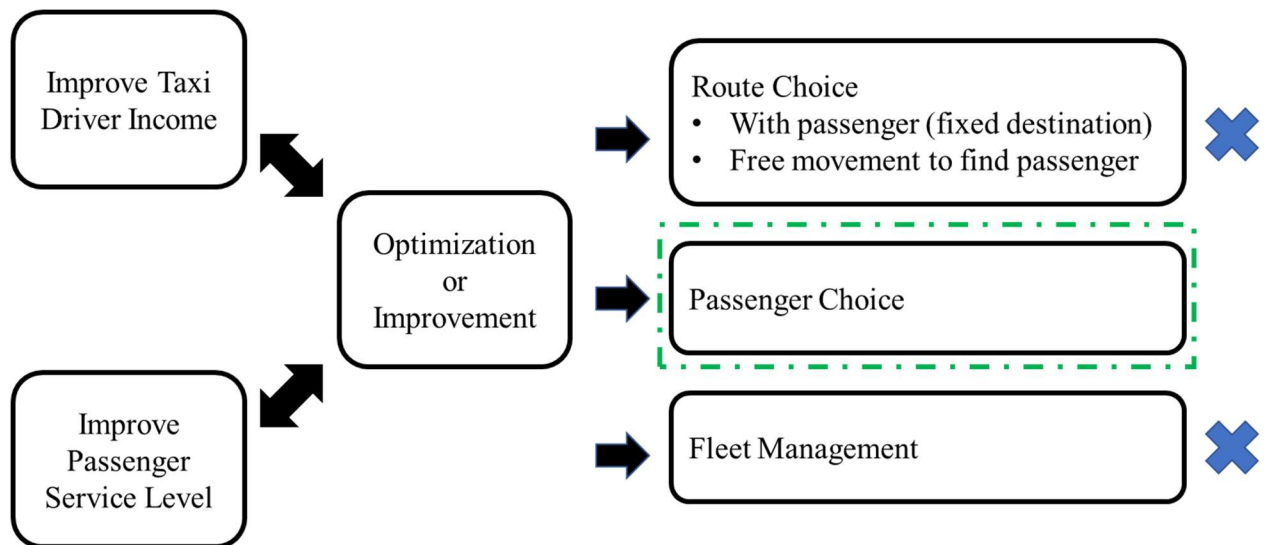


**Figure 4.1: Taxi operation optimization**

For the case of optimization different scenario could be considered. A general model for the optimized or the improved system is shown in Figure 4.1. The first scenario is how the driver pick up the passenger. In this scenario, driver have the ability to choose which passenger to pick and which not to pick. The reason behind of having driver choose the passenger is that the refusal rate would decrease drastically. This does not guarantee that all taxi driver is willing to server the customer. However, if certain number of taxi are not willing to server the customer, there will exist other group of vacant taxi driver that are willing to serve the customer making the demand and

supply theoretically in the state of equilibrium unless demand out run the supply. The second scenario is the routing strategies for the taxi driver. Choosing the optimum route also plays the vital role how taxi driver could efficient run the service with minimizing operational cost.

In addition, optimization itself could be conducted at multiple choices, such as ‘Route Choice’ optimization where optimized route for taxi with and without passenger could be established. Next is the ‘Passenger Choice’ optimization where the driver has the means to choose the better passenger out of all the available passengers. Lastly the fleet of taxi could be managed entirely for the optimum operation. Figure 4.2 shows the available optimization or improvement choice that could be implemented for the better operation. However, the scope here is limited to passenger choice optimization only



**Figure 4.2: Optimization or improvement choice**

#### 4.2 Optimization Policy

Optimization of the taxi driver working behavior was introduced to improve the driver income as well as improve the passenger service level. In order to quantitatively understand the optimized taxi operation indicator explaining the optimized service need to be defined properly. The indicator would essentially distinguish between the normal business as usual and the improved model. Three different indicators were selected for the purpose which are shown as follows.

### Indicator for optimization

- Reduce passenger waiting time
- Reduce travel distance without passenger
- Increase number of passenger trip of the taxi drivers

### Derived indicator for optimization

- Improve taxi driver daily income

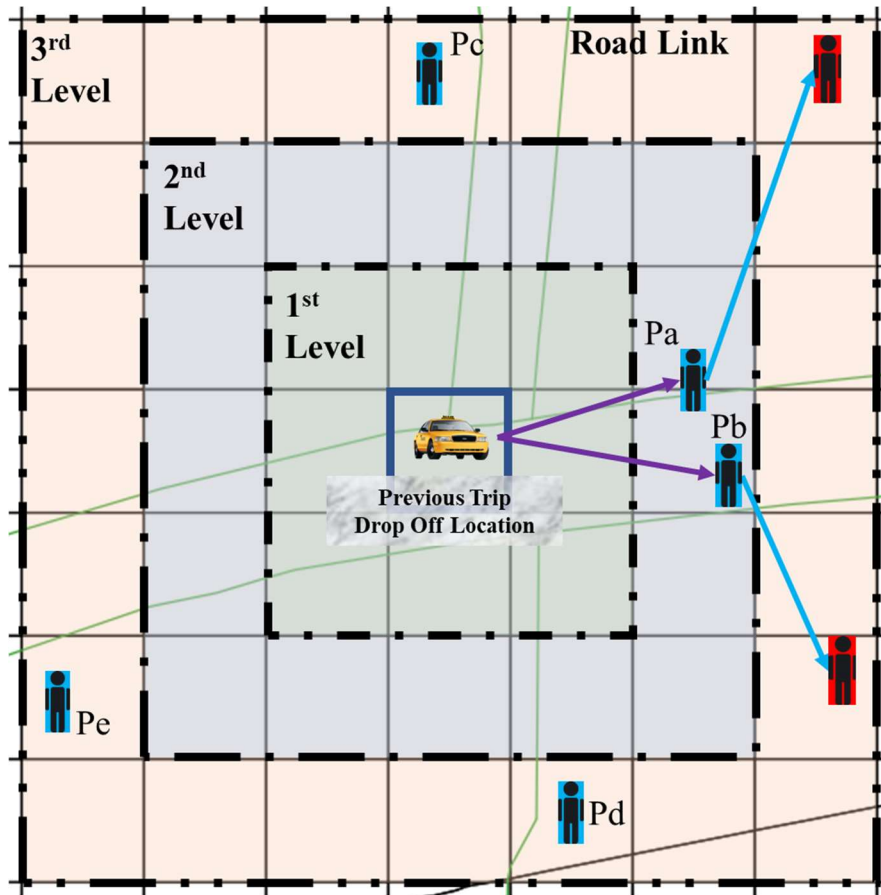
Assumption was made for optimization or the improvement model that the taxi driver has application that would search and give information about potential passenger within the given search space. Driver would choose the passenger that would be better in terms of profit i.e. choosing closer passenger as well as the passenger that can give better income. The optimized model was similar with the on-demand taxi service but with recommendation to which passenger to pick up.

### 4.3 Passenger Search Policy

Optimization or improvement of the driver behavior was introduced based on passenger choice. Given a taxi agent in a grid cell 'g' at a given time 't'. Agent starts to search the passenger in and around the grid. Passenger search space of 3x3 grid cell. Each grid cell of 500 meter. If there are no passenger, the search space is increased to 5x5 grid cell and so on. For each increment search time is added during the simulation. As the passenger is chosen it is removed from the available passenger demand so that another taxi cannot select that passenger. When multiple passenger is available within the search space the passenger is chosen getting high profit with low cost function. Passenger Selection is done based on simple cost function as shown in Equation 4.1.

$$Cost\ Function = \frac{Pick\ up\ distance}{Trip\ distance} \dots\dots\dots 4.1$$

Where, *Pick up distance* is the distance from the taxi agent to the passenger origin and *Trip distance* is the passenger trip distance from the passenger origin to the passenger destination. Based on the cost function lower the cost function better would be the recommendation for the taxi driver.



**Figure 4.3: Passenger Search Policy**

As shown, Figure 4.3 shows an example for the passenger search policy. For a given taxi present at a grid was considered as the previous trip drop off location. Hence, the state of the taxi as defined by taxi modeling state was the free movement taxi. However, in optimization model policy for searching of the passenger was altered as compared to the business as usual simulation. In optimized or the improved model, the taxi starts to search passenger at different search levels. In a given example taxi search passenger at 1<sup>st</sup> level of 3x3 grid. As there are no passenger at the 1<sup>st</sup> level, the search space was increased to 5x5 grid. For this search level there were two available

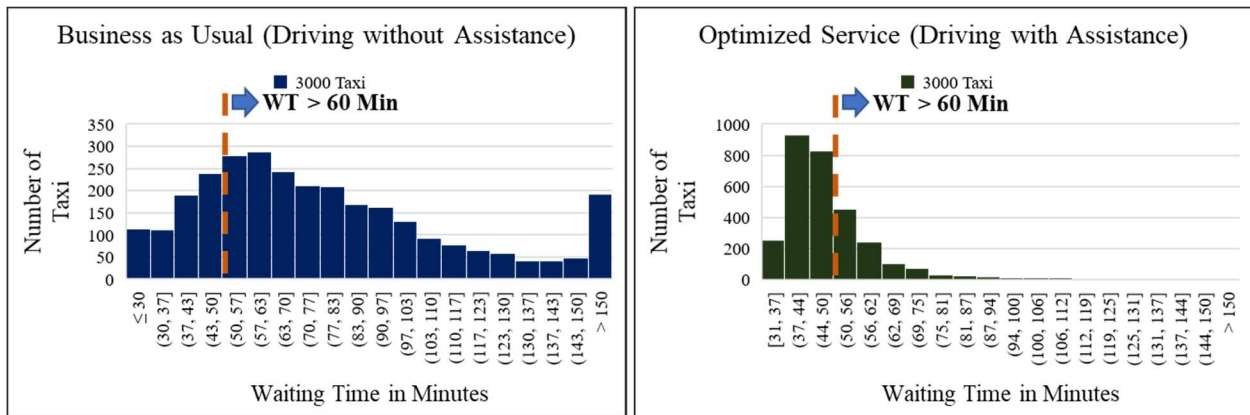
passengers Pa and Pb. However, passenger Pa was closer to the taxi agent as compared to passenger Pb. In addition, passenger Pa trip distance was longer as compared to passenger Pb. With this available information evaluated for based on the cost function, passenger Pa would have lower cost value than the passenger Pb and hence the choice of passenger for the taxi agent would be Pa. This simple optimization or the improved method of passenger search choice was implemented. The optimized model was then evaluated with business as usual model for different number of taxi agent which were 3000 taxi agents, 5000 taxi agents and 10000 taxi agents.

#### **4.4 Model Evaluation**

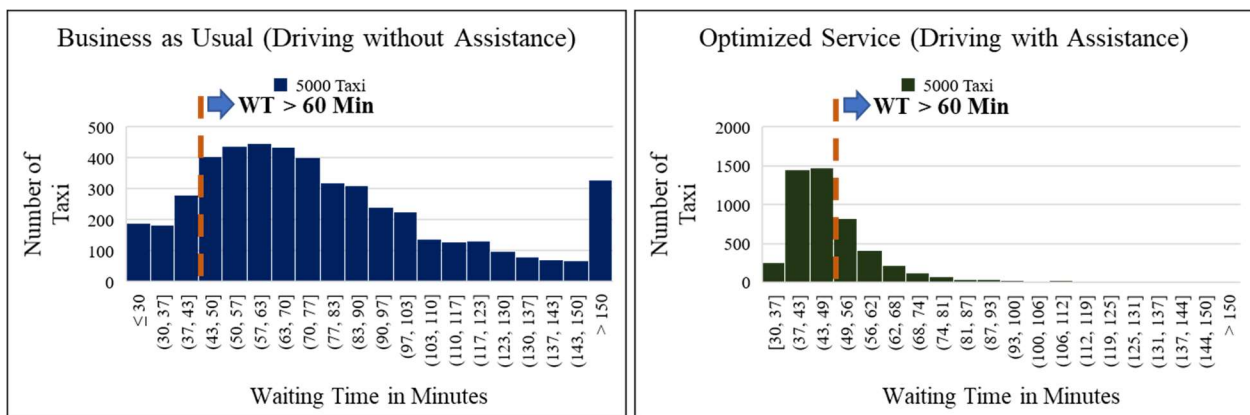
Evaluation of the model is an important aspect for determining the effectiveness of the implemented model. Three different number of taxi agents (3000 taxi agents, 5000 taxi agents and 10000 taxi agents) were simulated for business as usual and optimized or improved simulation. Evaluation were conducted for three indicators and one derived indicator which were passenger waiting time, distance traveled without passenger, driver's daily income and driver's total passenger trip per day.

##### **4.4.1 Passenger Waiting Time Evaluation**

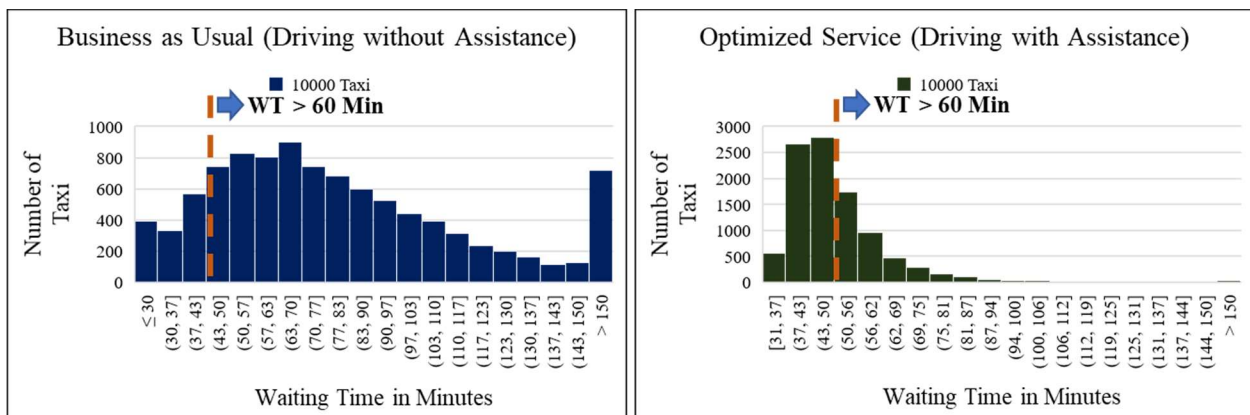
The evaluation between the business as usual and optimized model was done for the passenger waiting time evaluation. Figure 4.4-4.6 shows the evaluation for three different taxi agent sizes. For the case of 3000 taxi agents in Figure 4.4, there were many taxis had the waiting time for passenger more than 60 minutes with few taxi less than 60 minutes waiting time. As compared to the optimized simulation for the same number of agents, the number of taxi with waiting time to get next passenger was reduced significantly. Similarly Figure 4.5 and Figure 4.6 shows the waiting time to get next passenger for taxi agent 5000 and agent 10000. Result or the evaluation was similar with business as usual simulation with many taxi having higher waiting time as compared to the optimized simulation. When mentioning about the waiting time, if the waiting time to get the next passenger is higher it will automatically reduce the number of passenger the taxi driver could have served reducing its income. Hence, as the waiting time in the optimized simulation model was reduced, taxi operation was deemed as improved.



**Figure 4.4: Passenger waiting time comparison (3000 Taxi Agents)**



**Figure 4.5: Passenger waiting time comparison (5000 Taxi Agents)**

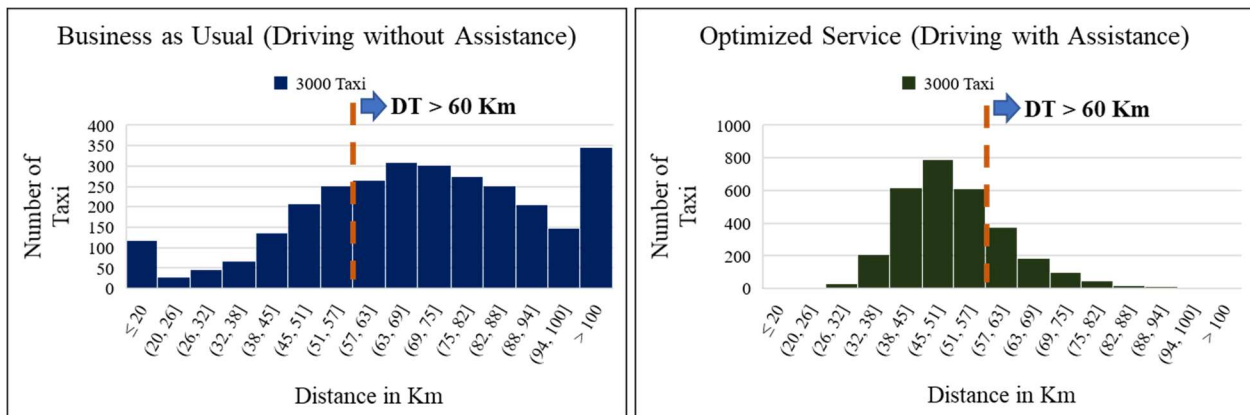


**Figure 4.6: Passenger waiting time comparison (10000 Taxi Agents)**

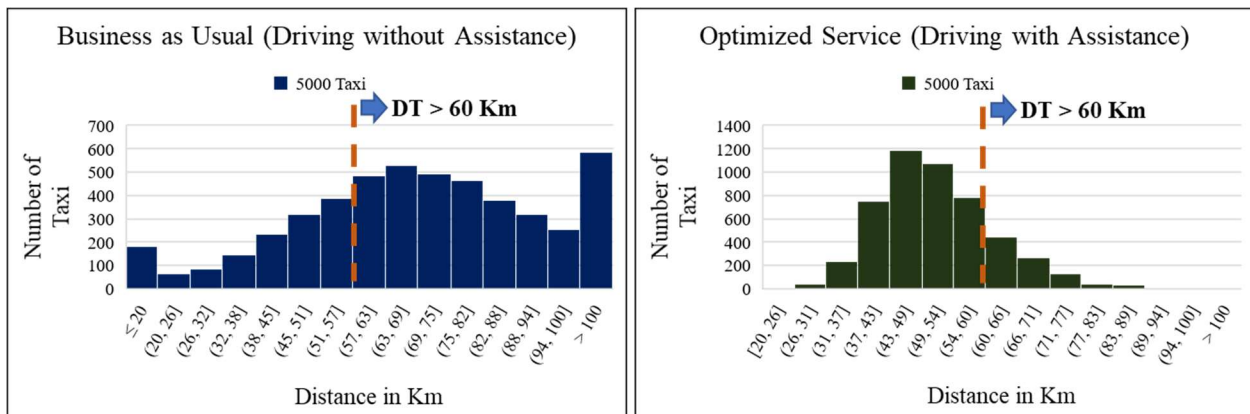


#### 4.4.2 Distance Travel Without Passenger Evaluation

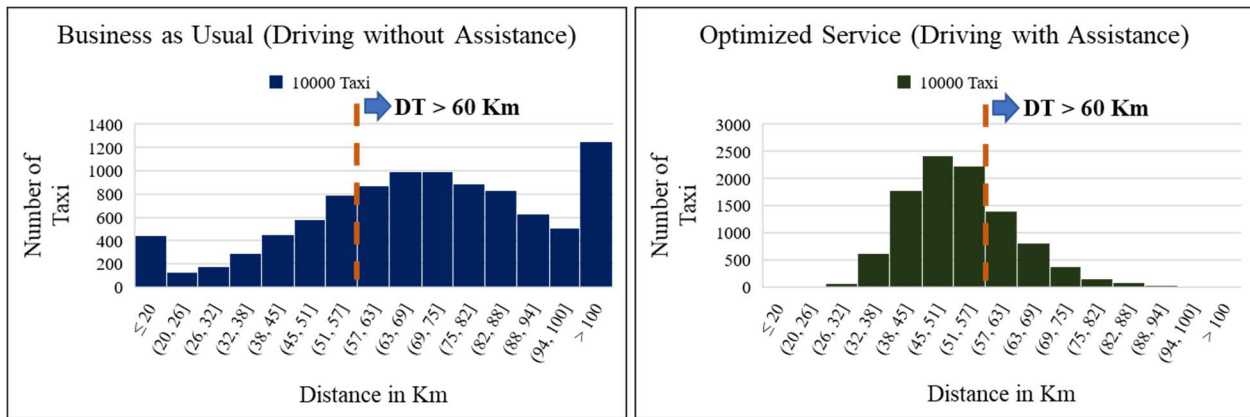
The evaluation between the business as usual and optimized model was done for the distance travelled by the taxi without any passenger. Figure 4.7-4.9 shows the evaluation for three different taxi agent sizes. For the case of 3000 taxi agents in Figure 4.7, the number of taxis driving more than 60 km daily without passenger was large in size in the business as usual simulation model. In this regard, the number of taxi driving more than 60 km daily without passenger was significantly reduced in the optimized model. Similar result trend was established for the taxi agent with 5000 taxis and 10000 taxis as shown in Figure 4.8 and Figure 4.9. Business as usual simulation showed significant high number of taxi driving with more than 60 km without passenger as compared to optimized model. When mentioning about the distance travel without passenger, lower the better as optimized model showed improvement in taxi operation for this model indicator.



**Figure 4.7: Daily distance travel without passenger comparison (3000 Taxi Agents)**



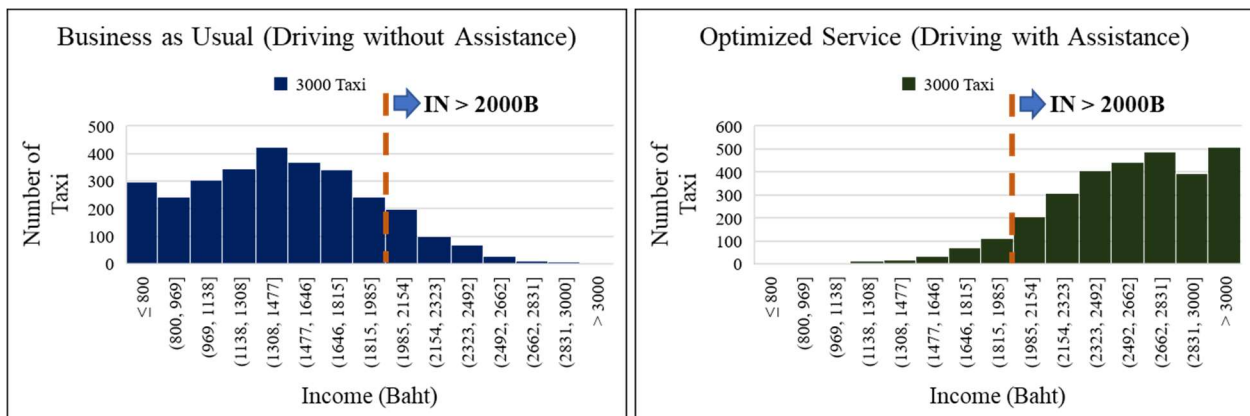
**Figure 4.8: Daily distance travel without passenger comparison (5000 Taxi Agents)**



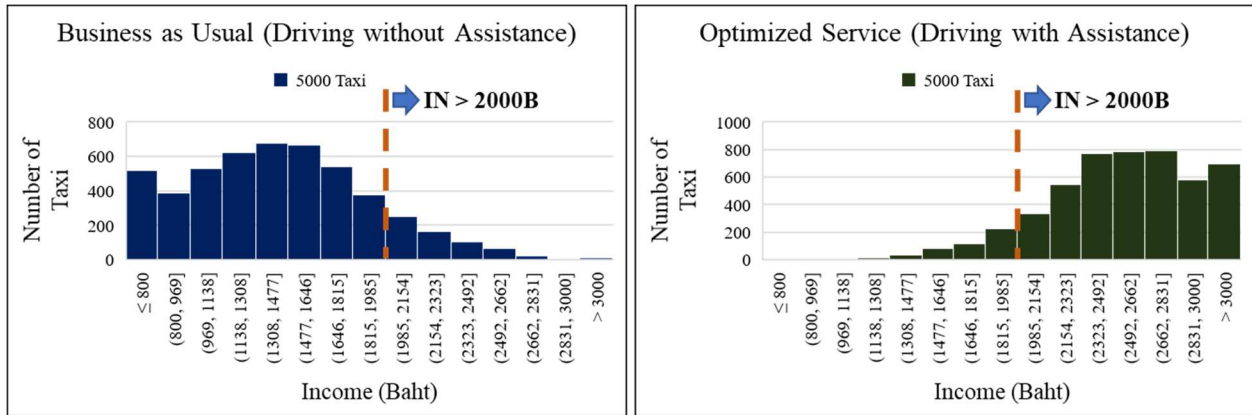
**Figure 4.9: Daily distance travel without passenger comparison (10000 Taxi Agents)**

### 4.4.3 Driver's Income Evaluation

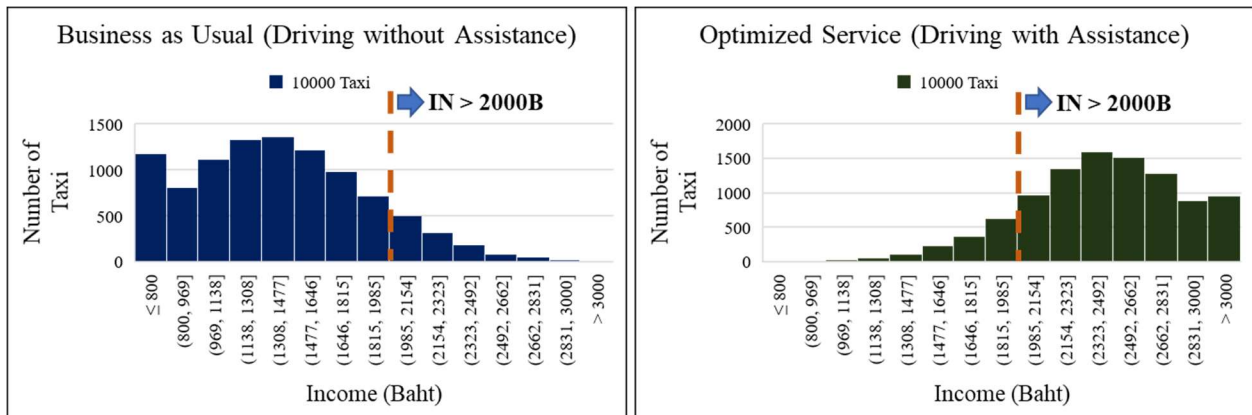
The evaluation between the business as usual and optimized model was done for the driver's daily income. Figure 4.10-4.12 shows the evaluation for three different taxi agent sizes. For the case of 3000 taxi agents in Figure 4.10, the number of taxis working with income less than 2000 baht was significantly less. In this regard, the number of taxi working with less than 2000 baht was significantly reduced in the optimized model. Similar result trend was established for the taxi agent with 5000 taxis and 10000 taxis as shown in Figure 4.11 and Figure 4.12. When mentioning about the taxi driver income the optimized model showed improvement in taxi operation significantly as many taxis were getting income raised which was the sole purposed for the model.



**Figure 4.10: Driver's daily income comparison (3000 Taxi Agents)**



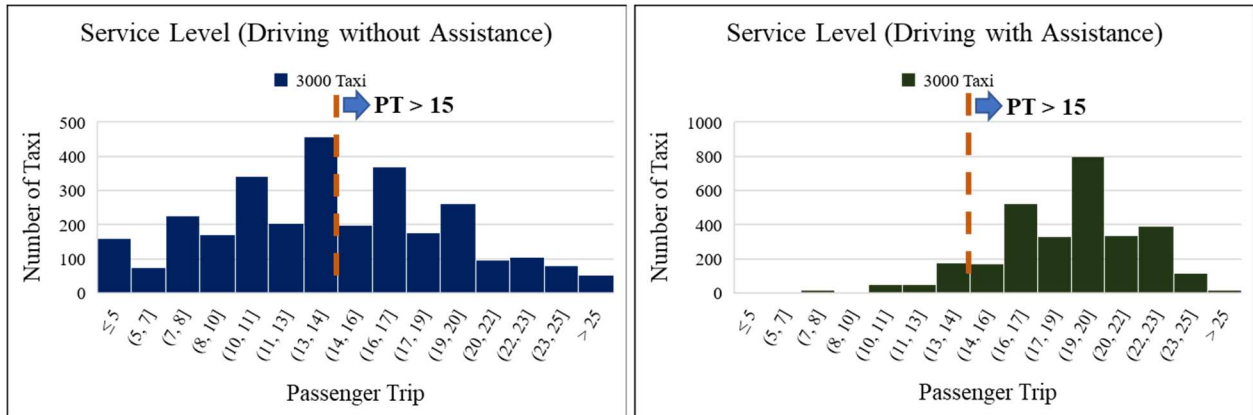
**Figure 4.11: Driver's daily income comparison (5000 Taxi Agents)**



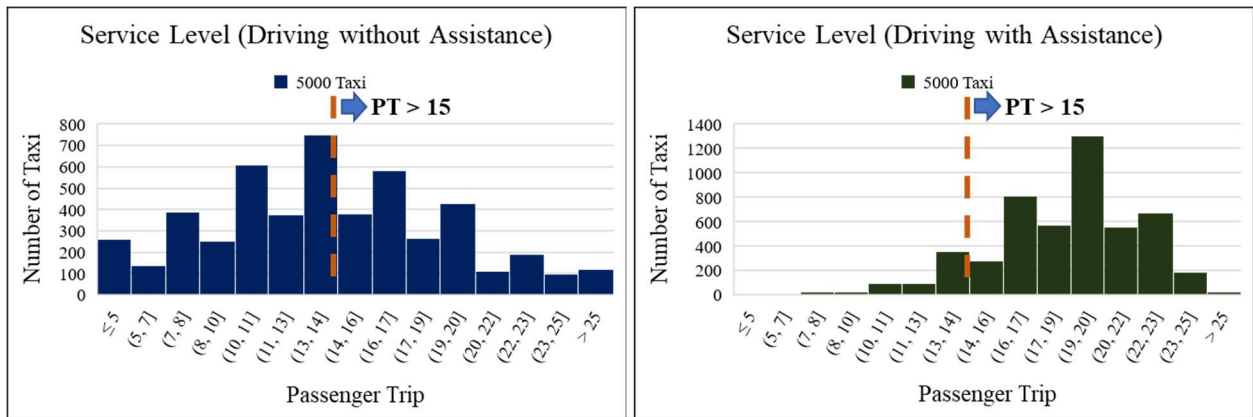
**Figure 4.12: Driver daily income comparison (10000 Taxi Agents)**

#### 4.4.4 Passenger Service Level Evaluation

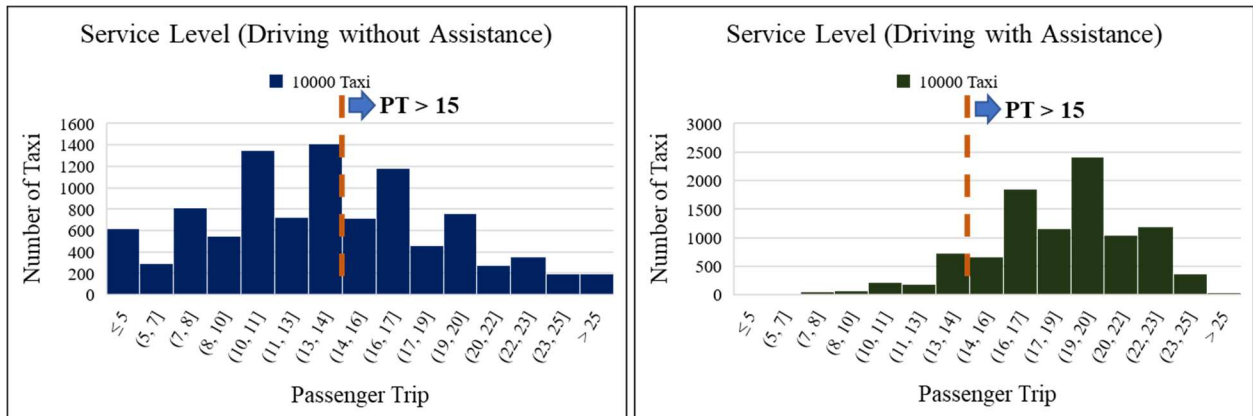
The evaluation between the business as usual and optimized model was done for the driver's daily passenger trip. Figure 4.13-4.14 shows the evaluation for three different taxi agent sizes. For the case of 3000 taxi agents in Figure 4.13, the number of taxis having passenger trip was significantly less. In this regard, the number of taxi with passenger trip was significantly increased in the optimized model. Similar result trend was established for the taxi agent with 5000 taxis and 10000 taxis as shown in Figure 4.15 and Figure 4.16. When mentioning about the taxi driver passenger trip the optimized model showed improvement in taxi operation significantly as many taxis were getting more number trip. The result was the improvement in the taxi service level as getting large number of passenger directly co-relates to reduction of the rejection rate of the passenger.



**Figure 4.13: Taxi passenger trip per day comparison (3000 Taxi Agents)**



**Figure 4.14: Taxi passenger trip per day comparison (5000 Taxi Agents)**



**Figure 4.15: Taxi passenger trip per day comparison (10000 Taxi Agents)**

Optimization of the taxi behavior is an important aspect for improving the taxi driver's profit up to an extent. Even with simple optimization that focus on reducing waiting time and increasing

trip could make help improve profit. Introducing simple improvement method as such selecting passenger through smart phone application could benefit both driver's income as well as improve the passenger service level.

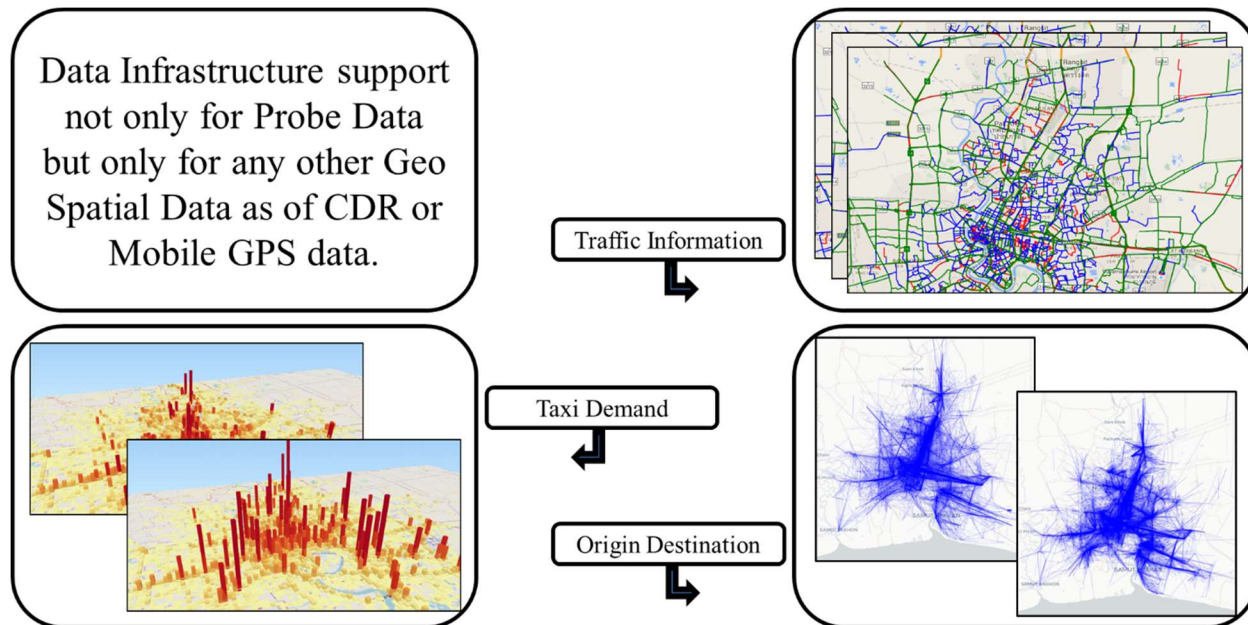
## CHAPTER 5

### CONCLUSION

5.

#### 5.1 Data Infrastructure

The development of the data infrastructure management system, for handling the big data facilitates significantly for working with the probe vehicle data. The system not only supports the probe taxi data but supports any type of big geo spatial data analysis whether it is from CDR data or mobile GPS data. An advantage of having a data infrastructure is shown in Figure 5.1.



**Figure 5.1: Advantage of data infrastructure**

Map matching is an important preliminary data analysis process in any spatial data analysis. The result obtained from map matching is a labelled GPS data set which is based on a OSM road network. Many matching algorithms are available, but choosing the best algorithm depends on kind of data set we have. More importantly, sensitivity of these map matching accuracy depends upon various factors such as road link types, GPS data points as well as the sampling interval of the GPS points. Varying one factor could change the overall accuracy of the whole map matching system. In addition, when dealing with a big spatial temporal data set, time factor may also play

important role. A large-scale computing platform as such distributed computing system would be an option which allows faster computational performance and is well suited with horizontal scaling for enhancing processing power even further.

With map-matched probe data set available that are operational and running throughout the cities, spatial and temporal information can be an asset for governing various urban management. As (Leduc 2008) indicated, accurate data from the probe vehicle can lead to a improvements in many area and services such as “Congestion reduction”, “Improve origin-destination matrices for commuter plans”, “Traffic queue detection”, “Improve incident management”, “Improve vehicle fleet management”, “Shorten driving time with optimization”, “Reducing fuel consumption and thus reducing CO<sub>2</sub> emission” etc. Application from the accurate probe data are immense of which benefits can be utilized by both urban planner as well as drivers. In addition, utilization of open data as of open street map road network, the technology could be replicated and implemented for any other country where open street map road network is readily available. Furthermore, with the open data of road network, user can customize or enhance the data set as per their requirement and need. Map matching methodology however, can be improved by introducing factors for better accuracy for various road link segments.

## **5.2 Taxi Simulation Modeling**

Modeling of taxi service is an important aspect of understanding the behavior of taxi service level in the city. A data driven agent-based simulation model provided a way to simulated taxi behaviors in a large-scale urban area with the taxi probe vehicle data. Analysis of the taxi agent simulation showed a significant similarity with the real taxi data, indicating that the simulated result could keep the real nature of taxi service behavior. In the agent-based modeling presented here, taxi service modeling was categorized based on weekday and weekend. Nevertheless, with the increasing utilization of GPS probe data, modeling of service can be made by adding other entities, such as daily variation, monthly variation, etc. More importantly, such simulation can help understand and predict the effect of having a large number of taxis in the spatial and temporal domain with low demand, and vice versa. Understanding such taxi behavior in the city can

significantly help managing and dispatching the fleet of the taxi that can make monetary profit for the drivers.

The limitation of the current agent-based model is that the current agent-based model system utilizes an offline learning method, which possesses constraints in terms of time and resources when required to learn from a high-speed streaming dataset. The offline learning method, despite having many use cases, possess a limitation in regards how it can handle new datasets. In such cases, the model needs to be improved that would help accommodate learning from high-speed steaming data, as proposed in (Luís Moreira-Matias et al. 2016), where the OD matrix constantly evolves as time progresses, with the addition of new datasets and removal of the outdated datasets. The model can further be improved in terms of free movement of vacant taxis, with regard to replacing movement directed by direction angle to searching the next road network node at each time interval. Furthermore, current agent-based modeling could describe the taxi behavior with a trip time of 2 h and trip distance of 100 km. Though such trips accounted for about 98% of the total trip, the model could be further improved to encapsulate both short and long trips, regarding both time and distance. In addition, some of the simulation indicator does poses some level of variance between the simulated and real behavior which may have caused due to homogeneity in the simulation model. Driver's skill factor could be added to introduce the heterogeneity among the taxi drivers that influences the pick-up success of a given taxi. Routing is done using shortest path algorithm which caused underestimation of simulated data in terms of travel distance and subsequently the income for taxi driver.

### **5.3 Taxi Optimization Modeling**

Optimization of the taxi behavior is an important aspect for improving the taxi driver's profit up to an extent. Even with simple optimization that focus on reducing waiting time and increasing trip could make help improve profit. Introducing simple improvement method as such selecting passenger through smart phone application could benefit both driver's income as well as improve the passenger service level. Optimization model provided recommendation to taxi driver to pick up the better passenger among different passengers. The result was taxi driver could make more overall income as well as the passenger service level was improved as rejection of the passenger



was reduced. As the evaluation between the business as usual simulation and the optimized or improved simulation, the waiting time to get the next passenger for the optimized simulation has been significantly reduced. Moreover, the daily distance travelled by the taxi agents is also reduced in the optimized model. Importantly taxi agents were able to make more number of trip in each daily drastically improving the income to more than 2000 baht. All this suggest that with simple change in the passenger search strategy improvement on the overall taxi operation could be achieved. In addition, overall optimizing method could be optimizing route for efficient taxi operation also plays an important role determining how much profit the driver can make by reducing the operation cost on fuel as well as its maintenance. Lastly, taxi optimization model could be improved by introducing route choice behavior along with passenger selection choice with proper fleet management that could be introduced in the model by comparing the relationship between supply and demand, for efficient driving.

## APPENDIX A

### MAP MATCHING TEST CASE

#### TEST CASES

- 1) Normal (single lane) road link (**Case 1**)
- 2) Multiple lane road link (**Case 2**)
- 3) Simple road intersection (**Case 3**)
- 4) Complex road intersection (**Case 4**)
- 5) Elevated road link (**Case 5**)
- 6) Large gap between GPS points at large time interval (**Case 6**)
- 7) Random jump of GPS points at short time interval (**Case 7**)

IMEI	Latitude	Longitude	Selected Link id	No. 1	No. 2	No. 3	No. 4	No. 5	No. 6	No. 7
10000023	14.2209	100.69523	200000183142		*			*		
10000023	14.2176	100.68918	200000183136		*			*		
10000023	14.2146	100.6834	200000183136		*			*		
10000023	14.2113	100.67738	200000183135		*			*		
10000023	14.2093	100.67355	200000183135		*			*		
10000023	14.2047	100.6651	200000183026		*			*		
10000023	14.2028	100.66162	200000183026		*			*		
10000023	14.2008	100.65792	200000183128		*			*		
10000023	14.1992	100.65473	200000183128		*			*		
10000023	14.1972	100.651	200000183129		*			*		
10000023	14.1949	100.64695	200000183129		*			*		
10000023	14.1929	100.64325	200000183129		*			*		
10000023	14.1913	100.64033	200000183060		*			*		
10000023	14.1889	100.63753	200000183059		*		*	*		
10000023	14.184	100.63993	200000183061		*			*		
10000023	14.1783	100.64297	200000184806		*			*		

10000023	14.1717	100.64668	200000184808		*			*		
10000023	14.1645	100.65075	200000183252		*			*		
10000023	14.1573	100.65477	200000183252		*			*		
10000023	14.1502	100.6588	200000183252		*			*		
10000023	14.1426	100.66313	200000183252		*			*		
10000023	14.1272	100.67187	200000183252		*			*		
10000023	14.1196	100.67608	200000183252		*			*		
10000023	14.1127	100.67997	200000183252		*			*		
10000023	14.1054	100.68408	200000183252		*			*		
10000023	14.0982	100.68822	200000183252		*			*		
10000023	14.0918	100.69175	200000183252		*			*		
10000023	14.085	100.69487	200000185327		*			*		
10000023	14.0775	100.6957	200000185327		*			*		
10000023	14.0682	100.6966	200000185327		*			*		
10000023	14.0596	100.69743	200000185328		*			*		
10000023	14.0508	100.69823	200000185328		*			*		
10000023	14.0416	100.69827	200000185328		*			*		
10000023	14.0329	100.6983	200000185328		*			*		
10000023	14.0248	100.69827	200000185328		*			*		
10000023	14.0178	100.70063	200000001064		*			*		
10000023	14.0103	100.70397	200000001064		*			*		
10000023	13.994	100.71118	200000011097		*			*		
10000023	13.9863	100.7134	200000011097		*			*		
10000023	13.9831	100.7133	200000112288		*			*		
10000023	13.9783	100.71335	200000013607		*			*		
10000023	13.9706	100.7131	200000013607		*			*		
10000023	13.9624	100.71293	200000013607		*			*		
10000023	13.954	100.71278	200000013607		*			*		
10000023	13.9472	100.71203	200000013607		*			*		
10000023	13.9393	100.70997	200000011100		*			*		

10000023	13.9317	100.70793	200000011100		*			*		
10000023	13.9236	100.70585	200000011102		*			*		
10000023	13.9165	100.70322	200000011102		*			*		
10000023	13.9095	100.69962	200000003380		*			*		
10000023	13.9021	100.69578	200000003380		*		*	*		
10000023	13.8949	100.69208	200000003379		*			*		
10000023	13.8876	100.68822	200000003379		*			*		
10000023	13.8718	100.68008	200000003379		*			*		
10000023	13.8635	100.67702	200000003379		*			*		
10000023	13.855	100.67478	200000003407		*			*		
10000023	13.8473	100.67265	200000003407		*			*		
10000023	13.8399	100.6726	200000003406		*			*		
10000023	13.832	100.67418	200000003406		*			*		
10000023	13.8237	100.67472	200000003406		*			*		
10000023	13.8161	100.6766	200000003406		*			*		
10000023	13.8074	100.6789	200000003404		*			*		
10000023	13.8004	100.68078	200000003404		*			*		
10000023	13.7933	100.68272	200000013585		*			*		
10000023	13.786	100.68463	200000013585		*			*		
10000023	13.778	100.68722	200000013585		*			*		
10000023	13.7703	100.69133	200000003386		*			*		
10000023	13.7633	100.69523	200000003386		*			*		
10000023	13.7575	100.69857	200000013659		*			*		
10000023	13.7464	100.70275	200000003402		*			*		
10000023	13.7386	100.70288	200000013590		*			*		
10000023	13.7304	100.70275	200000013590		*		*	*		
10000023	13.7226	100.7009	200000013590		*			*		
10000023	13.714	100.70032	200000003391		*			*		
10000023	13.7052	100.69993	200000003391		*			*		
10000023	13.6964	100.69965	200000003391		*			*		

10000023	13.6899	100.69972	200000003391		*			*		
10000023	13.6899	100.69972	200000003391		*			*		
10000023	13.6886	100.69963	200000003391		*			*		
10000023	13.6828	100.69953	200000003391		*			*		
10000023	13.6674	100.69877	200000003391		*			*		
10000023	13.6606	100.69465	200000003391		*			*		
10000023	13.6543	100.69033	200000012255		*			*		
10000023	13.648	100.68585	200000011731		*		*	*		
10000023	13.6431	100.68212	200000026516		*			*		
10000023	13.6415	100.68093	200000026502		*			*		
10000023	13.6369	100.6774	200000026502		*			*		
10000023	13.6316	100.67307	200000026502		*			*		
10000023	13.625	100.66762	200000026502		*			*		
10000023	13.6188	100.66268	200000012021		*			*		
10000023	13.6112	100.65835	200000026510		*		*	*		
10000023	13.6046	100.65428	200000026509		*			*		
10000023	13.6006	100.64745	200000026509		*			*		
10000023	13.6011	100.6391	200000026509		*			*		
10000023	13.604	100.63127	200000026509		*			*		
10000023	13.6083	100.62487	200000026509		*			*		
10000023	13.6109	100.62232	200000026583		*		*			
10000023	13.6127	100.61923	200000026568		*		*			
10000023	13.6097	100.6173	200000122030		*		*			
10000023	13.6123	100.6188	200000002685		*		*			
10000023	13.6179	100.62193	200000002694		*					
10000023	13.6243	100.62512	200000041988		*					
10000023	13.6282	100.62572	200000002708		*	*				
10000023	13.6286	100.62412	200000050941		*					
10000023	13.6288	100.62427	200000050941		*					
10000023	13.6308	100.62085	200000050699		*					

10000023	13.631	100.6195	200000050818	*					
10000023	13.6289	100.61808	200000050818	*					
10000023	13.629	100.61658	200000050988	*					
10000023	13.6291	100.61638	200000050988	*					
10000023	13.6292	100.6159	200000050988	*					
10000023	13.6291	100.61643	200000050988	*					
10000023	13.6287	100.61795	200000050818	*					
10000023	13.6306	100.61925	200000050818	*					
10000023	13.6317	100.61918	200000051016	*					
10000023	13.6335	100.61583	200000050688	*					
10000023	13.6354	100.61235	200000002689	*					
10000023	13.6371	100.60913	200000002687	*					
10000023	13.6392	100.60627	200000050683	*					
10000023	13.642	100.60282	200000046743	*					
10000023	13.6445	100.59497	200000115775	*					
10000023	13.643	100.5944	200000115775	*					
10000023	13.643	100.5944	200000115775	*					
10000023	13.6431	100.59307	200000088529	*					
10000023	13.6442	100.58787	200000053663	*					
10000023	13.6454	100.5814	200000054195	*					
10000023	13.6465	100.576	200000036416	*					
10000023	13.6479	100.56918	200000054133	*					
10000023	13.6489	100.56415	200000137048	*					
10000023	13.6503	100.55762	200000137048	*					
10000023	13.6509	100.55153	200000180407	*					
10000023	13.6532	100.54565	200000042512	*					
10000023	13.6545	100.54117	200000021616	*		*			
10000023	13.6496	100.54138	200000021625	*		*			

## APPENDIX B

แบบสอบถามสำรวจข้อมูลแท็กซี่

### TAXI SURVEY QUESTIONNAIRE STAGE 1

โปรดช่วยฉันตอบคำถามลงในช่องว่าง โดยจะใช้เวลาประมาณ 5-10 นาที ข้อมูลต่างๆเหล่านี้จะนำมาใช้สำหรับการทำงานวิทยานิพนธ์ของฉัน ขอขอบคุณเป็นอย่างมากสำหรับข้อมูลเหล่านี้

Please help me fill up this form. It will take about 5 – 10 Minutes of your valuable time. I will be very grateful for your help to fill up this form for my research work. Thank you very much.

Survey Location:

1. ในระยะเวลา 1 ปี คุณขับรถ Taxi กี่เดือน? (How many months do you drive taxi?)

- ทั้งปี (All Year)       3-6 เดือน (3 - 6 Month)       น้อยกว่า 3 เดือน (Less than 3 Month)

2. ในหนึ่งวันคุณขับรถ Taxi กี่ชั่วโมง? (How many hours do you drive taxi in a day?)

- มากกว่า 12 ชั่วโมง (More than 12 Hour)       6-12 ชั่วโมง (6 - 12 Hour)  
 น้อยกว่า 6 ชั่วโมง (Less than 6 Hour)

3. รายได้ที่คุณได้รับคิดเป็นจำนวนเงินเท่าไรต่อวัน? (How much do you earn per day?)

- มากกว่า 1,500 บาท (More than 1500 Baht)       1,000 – 1,500 บาท (1000 - 1500 Baht)  
 500 – 1,000 บาท (500 - 1000 Baht)       น้อยกว่า 500 บาท (Less than 500 Baht)

4. คุณมีการเปลี่ยนผลัดกับคนอื่นหรือไม่ใน 1 วัน? (Do you share the taxi with other each day?)

- มี (Yes)       ไม่มี (No)

5. และถ้ามี มีคนทั้งหมดกี่คนในการเปลี่ยนกันขับภายใน 1 วัน? (If Yes with how many people do you share the taxi?)

- มากกว่า 4 คน (More than 4 People)       4 คน (4 people)  
 3 คน (3 people)       2 คน (2 People)

6. คุณเป็นเจ้าของรถ Taxi หรือเช่าจากบริษัท? (Do you own your own taxi or rent taxi from company?)

ใช่ เป็นเจ้าของรถ (Taxi Yes, own a taxi)

เช่าจากบริษัท (Rent from a company)

ถ้าเช่าจากบริษัท คุณเช่าเป็นรายวันหรือรายชั่วโมง (if rent, do you rent by day or hour base)

.....

ถ้าเช่าจากบริษัท คุณต้องจ่ายค่าเช่าและค่าประกันรถเท่าไร (If rent from company, how much do you have to pay for the company for rent and insurance?)

.....

7. คุณต้องจ่ายค่าน้ำมันเชื้อเพลิงและค่าบำรุงรักษารถเท่าไรต่อเดือน? (How much do you pay for fuel and maintenance of the taxi in a month?)

มากกว่า 2,000 บาท (More than 2000 Baht)

1000 – 2000 บาท (1000 – 2000 Baht)

น้อยกว่า 1,000 บาท (Less than 1000 Baht)

8. คุณขับรถเป็นระยะทางกี่กิโลเมตรภายใน 1 วัน? (How many about kilometer (in about) do you drive in a day?)

มากกว่า 200 กิโลเมตร (More than 200 km)

100 – 200 กิโลเมตร (100 – 200 km)

น้อยกว่า 100 กิโลเมตร (Less than 100 km)

9. ปกติคุณขับรถไปรับผู้โดยสารบริเวณไหน? (Where do you normally drive to get the passenger?)

ภายใน กทม. (Inside Central Bangkok)

นอกพื้นที่ กทม. (Outside Central Bangkok)

10. ปกติคุณวิ่งรถบริเวณพื้นที่ไหน? (Where is your working area?)

ตอนกลางวัน (day):.....

ตอนกลางคืน (night):.....



11. ในการหาผู้โดยสาร คุณมักจะรอคิว ตามจุดต่างๆ ใช่หรือไม่? (Do you prefer queue for passenger?)

ใช่ Yes  ไม่ใช่ No

ถ้าใช่ ที่ไหนที่คุณจอดรอคิว? If Yes, where do you normally queue?

.....

12. ง่ายหรือไม่ที่จะหาผู้โดยสาร? (Is it easy to find passenger?)

ใช่ (Yes)  ไม่ใช่ (No)

13. เวลาใดที่คุณสามารถหาผู้โดยสารได้ง่าย? (What time you can find passenger easily?)

0:00 – 3:00  3:00 – 6:00  
 6:00 – 9:00  9:00 – 12:00  
 12:00 – 15:00  15:00 – 18:00  
 18:00 – 21:00  21:00 – 24:00

14. ปกติคุณใช้เวลากี่นาทีถึงจะได้ผู้โดยสารใหม่ (How many minutes do you wait to get passenger?)

.....

15. จำนวนของผู้โดยสารมีความแตกต่างกันหรือไม่ระหว่าง วันทำงานปกติ วันหยุดเสาร์อาทิตย์ วันที่มีฝนตก และวันหยุดตามเทศกาลต่างๆ

ในพื้นที่ กทม. (Is there difference in taxi passenger number during week day/ week end/  
rainy day/ any event in Bangkok?)

ต่างกัน (Yes)  ไม่ต่างกัน (No)

ถ้าต่างกัน แล้วช่วงไหนที่มีจำนวนผู้โดยสารมากที่สุด? (If Yes, which period you get more passenger?)

โปรดเขียนตอบด้านล่างนี้

.....

16. คุณมีความคิดเห็นอย่างไรเกี่ยวกับ Grab Taxi ในประเทศไทย ไม่จำเป็นต้องตอบถ้าคุณไม่รู้จัก Grab Taxi

(What do you think about Grab Taxi in Thailand? Not necessary to answer)

- ดีสำหรับคนขับ Taxi (Good for taxi driver)
- ไม่ดีสำหรับคนขับ Taxi (Not good for taxi driver)

17. คุณปฏิเสธผู้โดยสารบางครั้งหรือไม่ (Do you reject passenger sometime? Not necessary to answer)

- ใช่ (Yes)                       ไม่ใช่ (No)

ถ้ามี ทำไมคุณถึงปฏิเสธผู้โดยสาร (If Yes, why?)

- ไม่ใช่เส้นทางที่คุณขับรับส่งผู้โดยสาร (Not my usual route to drive taxi)
- จุดหมายปลายทางของผู้โดยสารไกลเกินไป (Home location far from passenger destination)
- เป็นช่วงเวลาที่ต้องเปลี่ยนกะกับผู้ขับท่านอื่น (Time to change shift for another driver)
- ระยะทางไกลมาก (Too long destination route)
- ต้องไปเติมแก๊ส (Time to refuel gas)
- ทางที่ให้ไป อยู่คนละด้านกับที่วิ่งอยู่ (Different direction influence your decision)
- ทางที่ให้ไป รถติดมาก (High traffic density)

18. คุณเคยมีประสบการณ์ที่พบผู้โดยสารที่มีนิสัยไม่ดีบ้างหรือไม่? (Do you have any experience of bad passenger behaviors?)

- เคย (Yes)                       ไม่เคย (No)

ถ้ามี มีทั้งหมดกี่ครั้ง และเป็นเรื่องเกี่ยวกับอะไร โปรดเขียนตอบด้านล่าง (If Yes, how many times (about))

.....

19. บ่อยครั้งหรือไม่ที่คุณได้มีโอกาสได้ รับ/ส่ง ผู้โดยสารจากสนามบิน? How often do you pick/drop passenger from airport? (Little bit ambiguous question)

- หลายครั้ง (Many times)                       ไม่กี่ครั้ง (Few time)                       ไม่เคย (Not applicable)

20. คุณเคยใช้บริการหรือแอปพลิเคชันต่างๆ เกี่ยวกับเส้นทางที่ใช้ในการหาผู้โดยสารเพิ่มขึ้น เพื่อเพิ่มรายได้ให้กับคุณบ้างหรือไม่? (Would you use a service/application which optimize taxi route to get more passenger and increase profit?)

เคย (Yes)                       ไม่เคย (No)

ถ้าเคย บริการหรือแอปพลิเคชันตัวไหนที่คุณใช้ในการช่วยหาผู้โดยสาร If Yes, which application you use for find more passenger)

.....



## APPENDIX C



### แบบสอบถามสำรวจข้อมูลแท็กซี่ TAXI SURVEY QUESTIONNAIRE STAGE 2-3

โปรดช่วยตอบคำถามลงในช่องว่าง โดยจะใช้เวลาประมาณ 5-10 นาที ข้อมูลต่างๆเหล่านี้จะนำมาใช้สำหรับการทำงานวิทยานิพนธ์ (ไม่เกี่ยวข้องกับหน่วยงานราชการใดๆ) ขอขอบพระคุณเป็นอย่างมากสำหรับข้อมูลเหล่านี้ Please help me fill up this form. It will take about 5 – 10 Minutes of your valuable time. I will be very grateful for your help to fill up this form for my research work. Thank you very much.

**วัตถุประสงค์ (objective):** เพื่อศึกษากำไรให้บริการของรถแท็กซี่ในกรุงเทพฯ และปริมณฑล โดยเราได้ตั้งสมมติฐานเกี่ยวกับกำไรให้บริการของรถแท็กซี่ตามความเข้าใจพื้นฐานของเรา ดังนั้น การสำรวจข้อมูลนี้ จึงช่วยสนับสนุนข้อสมมติฐาน และปัจจัยสำหรับการศึกษาในลำดับต่อไป) The objective of this survey is to understand the operation of Taxi from the status of the Taxi Driver working in Bangkok and surrounding. We have developed some hypothesis about Taxi operation by our general understanding. This survey when conducted will help us validate or invalidate our hypothesis, which then will be used as an initial parameter for our research work that involves the Taxi Probe Data.)

Survey Location: .....

Time: .....

#### Section A: Personal Information

1. เพศ ? (Gender)

ชาย (Male)  หญิง (Female)

2. อายุ ? (Age)

< 20 ปี  20 - 30 ปี  30 - 40 ปี  40 - 50 ปี  > 50 ปี

3. ระดับการศึกษา (Education)

ประถมศึกษา (Primary School)

มัธยมศึกษา (High School)

ปริญญาตรี (Bachelor Degree)

สูงกว่าปริญญาตรี (Master Degree or higher)

4. พักอาศัยอยู่บริเวณ? (Where do you live?)

.....

**Section B: Taxi Working Information**

1. คุณขับแท็กซี่เป็นระยะเวลากี่ปี? (How long have you been driving taxi?)

< 2 ปี

2 - 5 ปี

5 - 10 ปี

10 - 20 ปี

> 20 ปี

2. โดยปกติคุณเริ่มต้นขับแท็กซี่จากที่ไหน? (Where do you start your day?)

บ้าน (Home)  อยู่แท็กซี่ (Taxi service center) โปรด

ระบุ.....

อื่นๆ (Others) โปรด

ระบุ.....

3. คุณเริ่มขับรถแท็กซี่เวลากี่โมง (What time do you start working?)

.....

4. คุณเลิกขับแท็กซี่เวลากี่โมง (What time do you finish your work?)

.....

5. คุณมีการเปลี่ยนกะกับคนอื่นหรือไม่ใน 1 วัน? (Do you share the taxi with other each day?)

มี (Yes)

ไม่มี (No)

6. ปกติคุณขับแท็กซี่ที่กลางวันหรือกลางคืน? (When do you normally drive a taxi?)

- กลางวัน (Day time)       กลางคืน (Night time)

7. ปกติคุณขับรถ รับ-ส่ง ผู้โดยสารบริเวณไหน? (Where is your service area?)

- กรุงเทพมหานคร (เขตพระนคร ดุสิต ป้อมปราบศัตรูพ่าย สัมพันธวงศ์ ดินแดง ห้วยขวาง ญาไท ราชเทวี และวังทองหลาง)
- กรุงเทพฯ (เขตปทุมวัน บางรัก สาทร บางคอแหลม ยานนาวา คลองเตย วัฒนา พระโขนง สวนหลวง และบางนา)
- กรุงเทพฯ (เขตจตุจักร บางซื่อ ลาดพร้าว หลักสี่ ดอนเมือง สายไหม และบางเขน)
- กรุงเทพฯ (เขตบางกะปิ สะพานสูง บึงกุ่ม คันนายาว ลาดกระบัง มีนบุรี หนองจอก คลองสามวาและประเวศ)
- กรุงเทพฯ (เขตธนบุรี คลองสาน จอมทอง บางกอกใหญ่ บางกอกน้อย บางพลัด ดลิ่งชันและทวีวัฒนา)
- กรุงเทพฯ (เขตภาษีเจริญ บางแค หนองแขม บางขุนเทียน บางบอน ราษฎร์บูรณะและทุ่งครุ)

8. ง่ายหรือไม่ที่จะหาผู้โดยสาร? (Is it easy to find passenger?)

- ใช่ (Yes)       ไม่ใช่ (No)

9. ที่ไหนที่สามารถหาผู้โดยสารได้ง่าย? (Where can you get passenger easily?)

.....

10. ที่ไหนที่คุณไม่อยากจะขับรถแท็กซี่ไป? (Where do you really not want to go?)

.....

ทำไม (Why).....

11. คุณขับแท็กซี่เป็นอาชีพหลักหรืออาชีพรอง

- อาชีพหลัก (Main job)       อาชีพรอง (Second job)

12. ในระยะเวลา 1 สัปดาห์ คุณขับรถ Taxi กี่วัน? (How many days do you drive taxi in a week?)

- ทุกวัน (Everyday)       5 วัน/สัปดาห์ (5 days/week)       น้อยกว่า 3 วัน/สัปดาห์ (Less than 3 days/week)

13. คุณขับรถเป็นระยะทางกี่กิโลเมตรภายใน 1 วัน? (How many about kilometer (in about) do you drive in a day?)

- มากกว่า 400 กิโลเมตร (> 400 km)       300 – 400 กิโลเมตร (300 – 400 km)  
 200 – 300 กิโลเมตร (200 – 300 km)       น้อยกว่า 200 กิโลเมตร (< 200 km)

14. คุณรับผู้โดยสารจำนวนกี่เที่ยวต่อวัน? (How many trips do you make per day?)

- มากกว่า 20 เที่ยว (> 20 trips)       15 – 20 เที่ยว (15 - 20 trips)  
 10 – 15 เที่ยว (10 - 15 trips)       น้อยกว่า 10 เที่ยว (< 10 trips)

15. ปกติคุณใช้เวลากี่นาทีถึงจะได้ผู้โดยสารใหม่ (How long do you wait to get passenger?)

..... นาที (minutes)

16. ในการหาผู้โดยสาร คุณมักจะรอคิว ตามจุดต่างๆ ใช่หรือไม่? (Do you prefer to queue for passenger?)

ใช่ (Yes) คุณจอรอคิวที่ไหน? (If Yes, where?)

.....

ไม่ใช่ (No)

17. เวลาใดที่คุณสามารถหาผู้โดยสารได้ง่าย? (What time you can find passenger easily?)

กรุณาใส่ตัวเลขจากมากไปน้อย 5 อันดับ โดยที่ 1 = มากที่สุด (Highest) และ 5 = น้อยที่สุด (Lowest)

- 0:00 – 3:00     3:00 – 6:00     6:00 – 9:00     9:00 – 12:00  
 12:00 – 15:00     15:00 – 18:00     18:00 – 21:00     21:00 – 24:00

18. เลือกวันที่มีจำนวนของผู้โดยสารจากมากไปน้อย? (The number of passenger in different day?)

กรุณาใส่ตัวเลขจากมากไปน้อย โดยที่ 1 = มากที่สุด (Highest) และ 4 = น้อยที่สุด (Lowest)

\_\_\_\_\_ วันทำงานปกติ (Weekday)

\_\_\_\_\_ วันหยุดเสาร์อาทิตย์ (Weekend)

\_\_\_\_\_ วันที่มีฝนตก (Rainy day)

\_\_\_\_\_ วันหยุดตามเทศกาลต่างๆ (Festival day / Holiday)

19. คุณอยากไปรับ-ส่ง ผู้โดยสารที่สนามบินหรือไม่? Do you prefer to go to airports?

ใช่ (Yes)

ไม่ (No) ทำไม (why) โปรดระบุ

.....

20. คุณได้ค่าโดยสารเฉลี่ยต่อเที่ยวเท่าไร? (How much is the average among you get in a trip?)

..... บาท/เที่ยว (Bath/trip)

21. ระยะทางเท่าไรที่คุณอยากไป? (What is your preference for driving?)

ระยะทางใกล้ น้อยกว่า 10 กิโลเมตร (Short distance < 10 Kilometers)

ระยะทางปานกลาง ระหว่าง 10 – 25 กิโลเมตร (Moderate distance 10 – 25 Kilometers)

ระยะทางไกล มากกว่า 25 กิโลเมตร (Long distance > 25 Kilometers)

22. คุณเคยมีประสบการณ์ที่พบผู้โดยสารที่มีนิสัยไม่ดีบ้างหรือไม่? (Do you have any experience of bad passenger behaviors?)

เคย (Yes)  ไม่เคย (No)

ถ้ามี ส่วนมากเป็นเรื่องเกี่ยวกับอะไร โปรดเขียนตอบด้านล่าง (If Yes, what is the usual issue?)

.....

23. คุณปฏิเสธผู้โดยสารหรือไม่ (Have you ever rejected passenger? Not necessary to answer)

ใช่ (Yes)  ไม่ใช่ (No)

ถ้ามี ทำไมคุณถึงปฏิเสธผู้โดยสาร (If Yes, please select the first five reason)

โปรดเลือก 5 อันดับ จากมากไปน้อย โดยที่ 1 = มากที่สุด (Most) และ 5 = น้อยที่สุด (Least)

\_\_\_\_\_ ไม่ใช่เส้นทางที่คุณเข้ารับส่งผู้โดยสาร (Not my usual route to drive taxi)

\_\_\_\_\_ จุดหมายปลายทางของผู้โดยสารไกลเกินไป (Home location far from passenger destination)

\_\_\_\_\_ เป็นช่วงเวลาที่ต้องเปลี่ยนกะกับผู้ขับท่านอื่น (Time to change shift for another driver)



\_\_\_\_\_ ระยะทางไกลมาก (Too long destination route)

\_\_\_\_\_ ต้องไปเติมแก๊ส (Time to refuel gas)

\_\_\_\_\_ ทางที่ให้ไป อยู่คนละด้านกับที่วิ่งอยู่ (Different direction influence your decision)

\_\_\_\_\_ ทางที่ให้ไป รถติดมาก (High traffic density)

\_\_\_\_\_ อื่นๆ (Others)

.....

### Section C: Expenses

24. คุณเป็นเจ้าของรถ Taxi หรือเช่าจากบริษัท? (Do you own your own taxi or rent taxi from company?)

ใช่ เป็นเจ้าของรถ (Taxi Yes, own a taxi)

เช่าจากบริษัท (Rent from a company)

ถ้าเช่าจากบริษัท คุณเช่าเป็นรายกะหรือรายวัน (if rent, do you rent by half-day or one-day)

รายกะ หรือ ครึ่งวัน (Half-day)  รายวัน (One-day)

ถ้าเช่าจากบริษัท คุณต้องจ่ายค่าเช่าและค่าประกันรถเท่าไร? (If rent from company, how much do you have to pay for the company for rent and insurance?)

.....

25. คุณต้องจ่ายค่าบำรุงรักษาเท่าไรต่อเดือน? (How much do you pay for maintenance of the taxi in a month?)

มากกว่า 2,000 บาท (> 2,000 Baht)

1,000 – 2,000 บาท (1,000 – 2,000 Baht)

น้อยกว่า 1,000 บาท (<1,000 Baht)

ไม่เสียค่าใช้จ่าย (No need to pay)

26. คุณต้องจ่ายค่าน้ำมันเชื้อเพลิงเท่าไรต่อวัน? (How much do you pay for fuel in a day?)

มากกว่า 400 บาท (> 400 Baht)  300 – 400 บาท (300 – 400 Baht)

200 – 300 บาท (200 – 300 Baht)  น้อยกว่า 200 บาท (< 200 Baht)

27. รายได้ที่คุณได้รับหลังจากหักค่าแก๊ส/น้ำมันคิดเป็นจำนวนเงินเท่าไรต่อวัน? (How much do you earn in a day? Net income)

มากกว่า 1,500 บาท (> 1500 Baht)       1,000 – 1,500 บาท (1000 - 1500 Baht)

500 – 1,000 บาท (500 - 1000 Baht)       น้อยกว่า 500 บาท (< 500 Baht)

#### Section D: Application

28. คุณเคยใช้บริการหรือแอปพลิเคชันต่างๆ เกี่ยวกับเส้นที่ใช้ในการหาผู้โดยสารเพิ่มขึ้น เพื่อเพิ่มรายได้ให้กับคุณบ้างหรือไม่? (Have you ever used any application a service/application to get more passengers and increase profit?)

เคย (Yes) โปรดระบุแอปพลิเคชัน (Application name).....

ไม่เคย (No)

ถ้ามีบริการหรือแอปพลิเคชันตัวไหนที่คุณใช้ในการช่วยหาผู้โดยสารคุณจะใช้หรือไม่ (Will you use application which we will develop that will optimize taxi route to get more passenger and increase profit?)

ใช่ (Yes)

ไม่ใช่ (No) โปรดระบุสาเหตุ (Why)

.....

29. คุณต้องเสียค่าใช้จ่าย สำหรับวิทยุ, GPS, หรืออินเทอร์เน็ต หรือไม่

ใช่ (Yes) เป็นเงินเท่าไร ..... บาท/เดือน       ไม่ใช่ (No)

30. คุณรู้จัก Grab Taxi หรือไม่? (Do you know about Grab Taxi or not?)

รู้จัก (Yes)       ไม่รู้จัก (No)

ถ้ารู้จัก คุณมีความคิดเห็นอย่างไรเกี่ยวกับ Grab Taxi ในประเทศไทย (If Yes, What do you think about Grab Taxi in Thailand?)

ดีสำหรับคนขับ Taxi (Good for taxi driver)

ไม่ดีสำหรับคนขับ Taxi (Not good for taxi driver)

หากคุณเป็นผู้ใช้ Grab Taxi คุณได้ลูกค้าจากการใช้แอปพลิเคชันนี้ กี่รอบต่อวัน?

- น้อยกว่า 5 รอบ (Less than 5 times)     5 - 10 รอบ (5 - 10 times)     มากกว่า 10 รอบ (Less than 10 times)

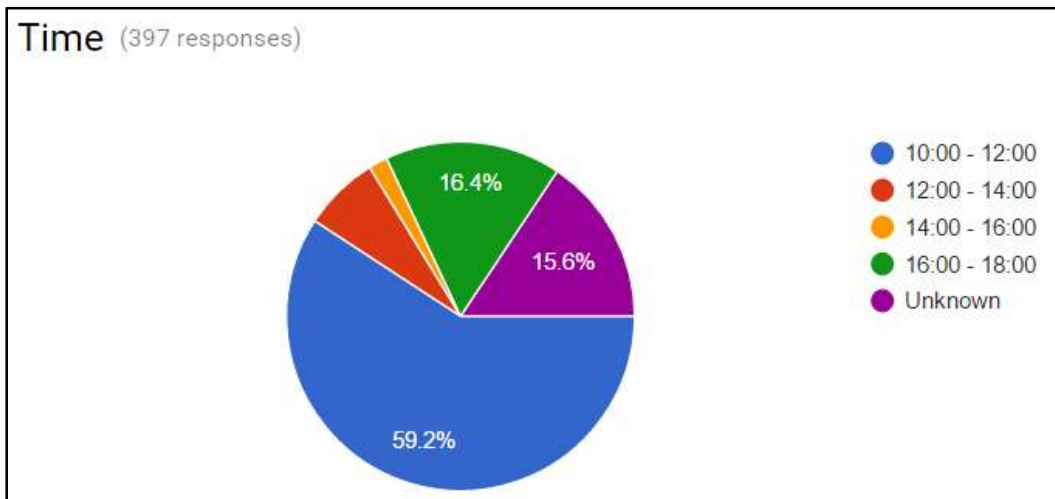
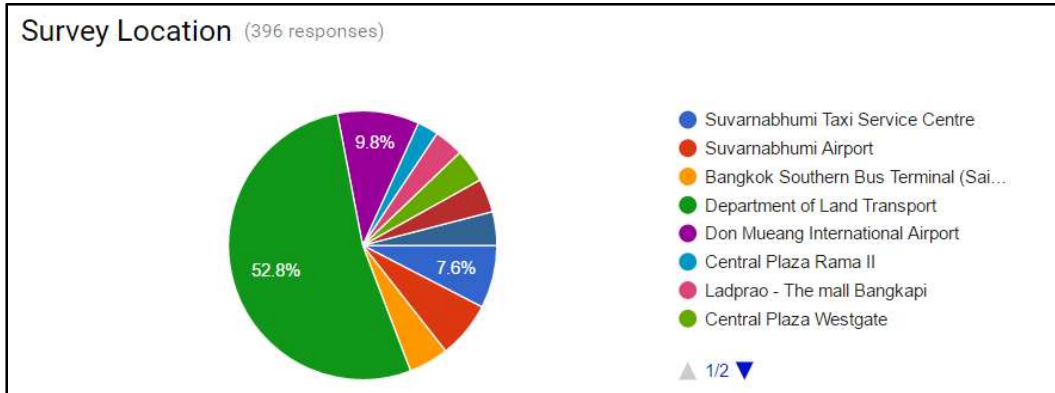
ช่วงเวลาไหนที่คุณได้ลูกค้ามากที่สุด?

กรุณาใส่ตัวเลขจากมากไปน้อย 5 อันดับ โดยที่ 1 = มากที่สุด (Highest) และ 5 = น้อยที่สุด (Lowest)

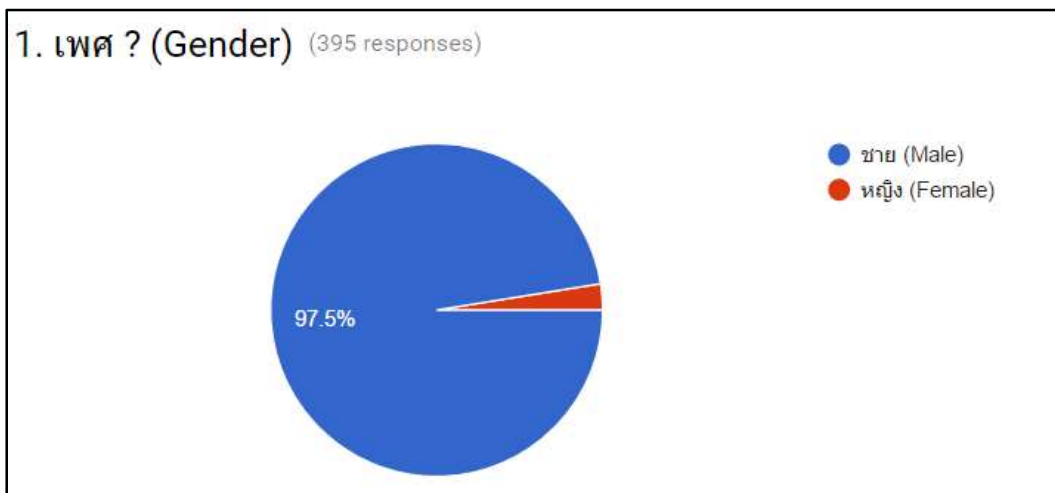
- 0:00 – 3:00     3:00 – 6:00     6:00 – 9:00     9:00 – 12:00  
 12:00 – 15:00     15:00 – 18:00     18:00 – 21:00     21:00 – 24:00

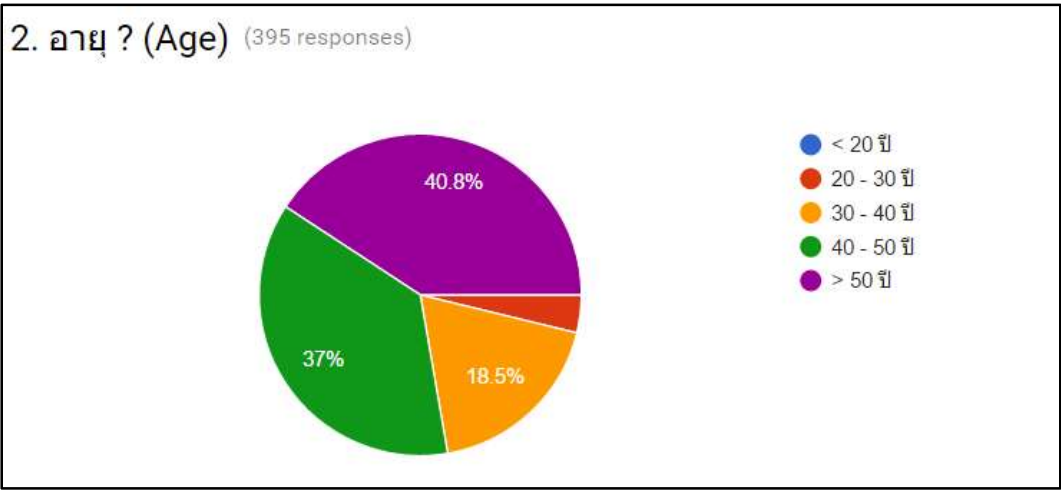
----- Thank you so much -----

## TAXI SURVEY QUESTIONNAIRE STAGE 2-3 RESPONSE

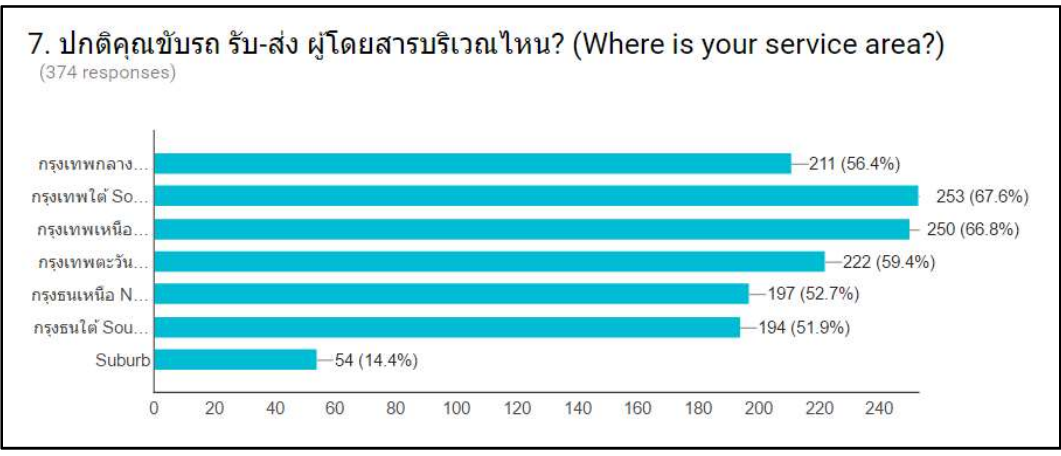
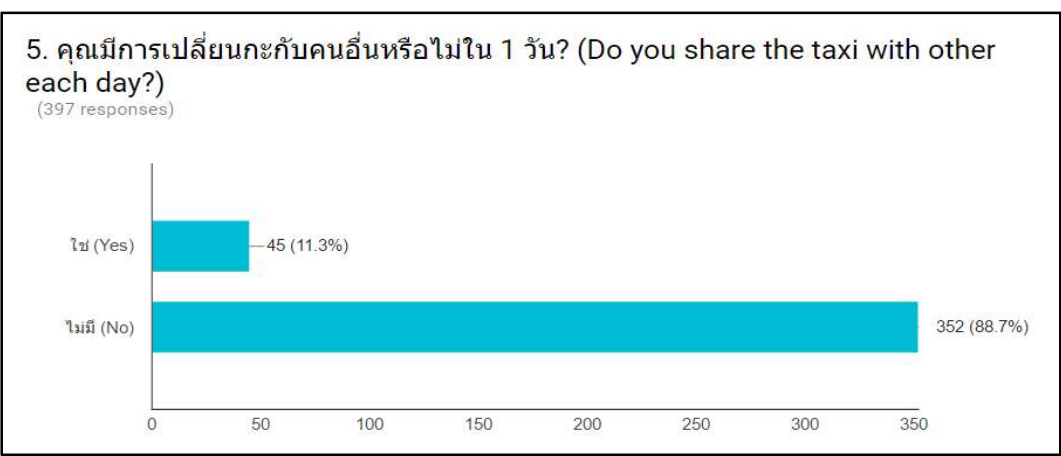


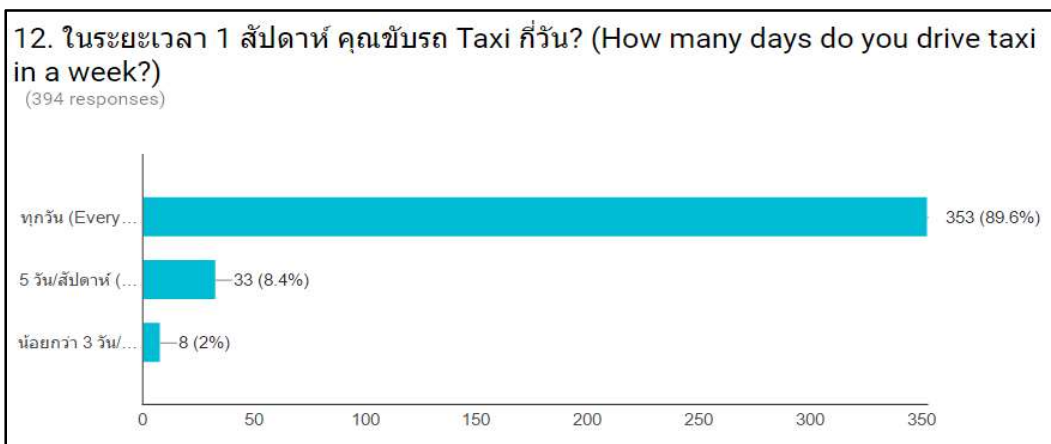
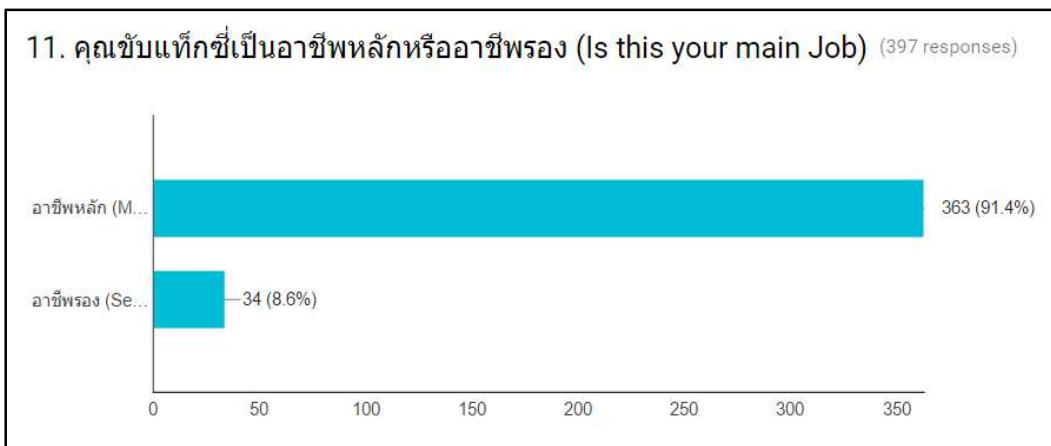
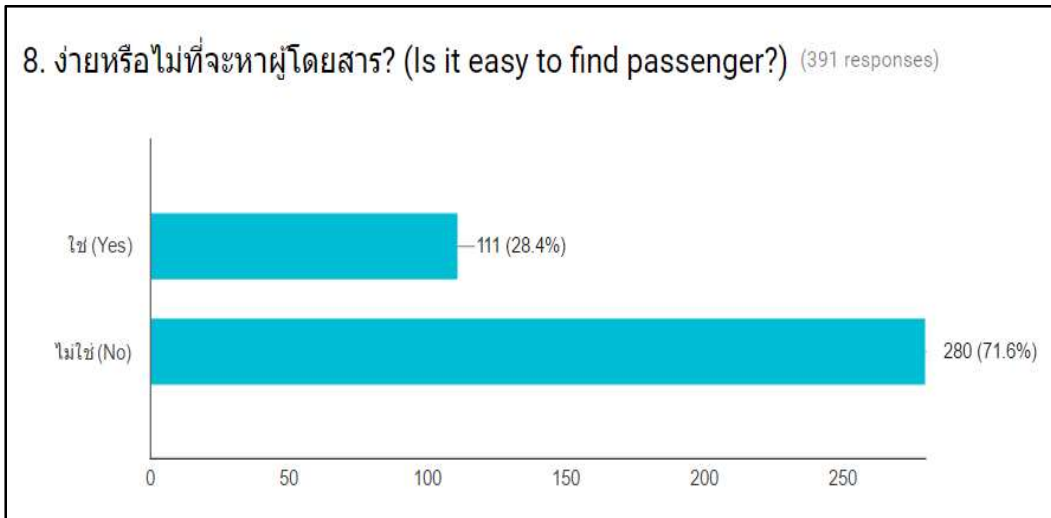
### Section A: Personal Information

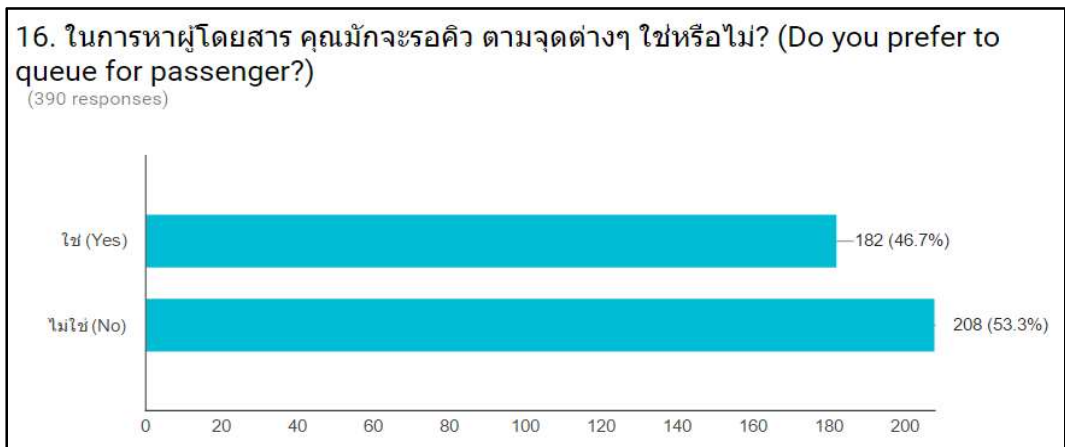
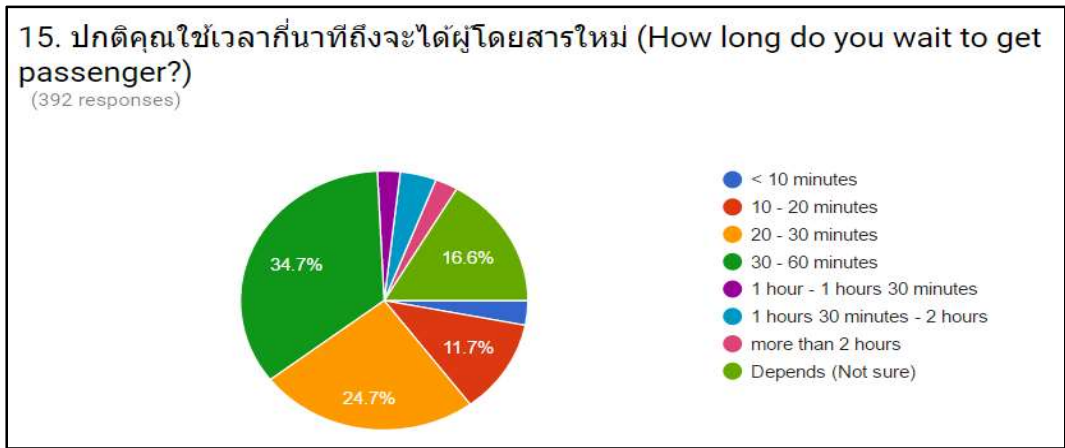
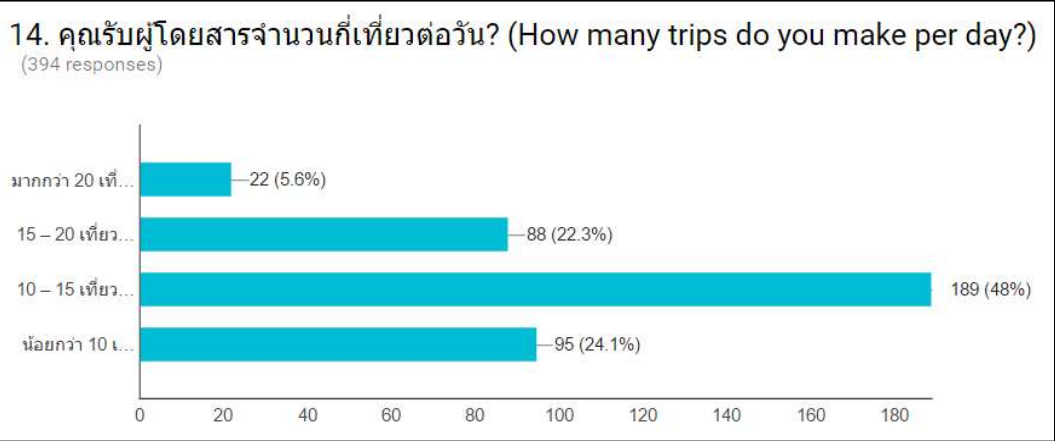


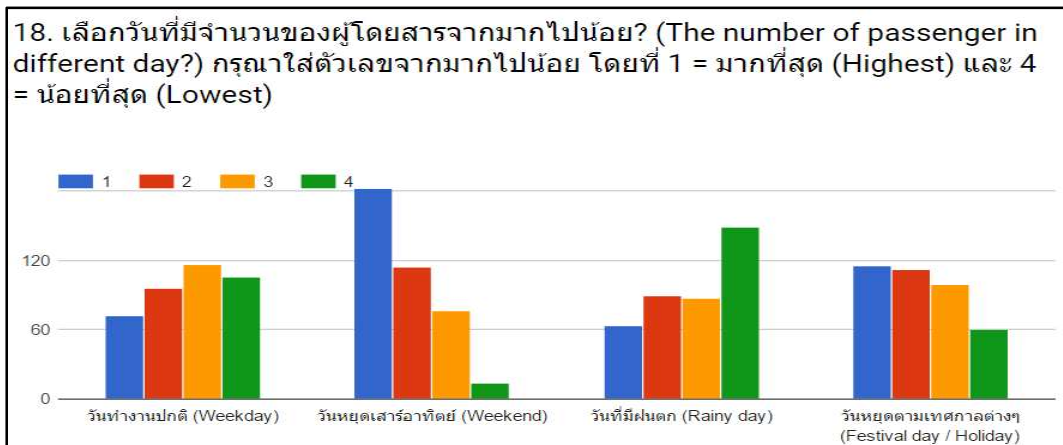
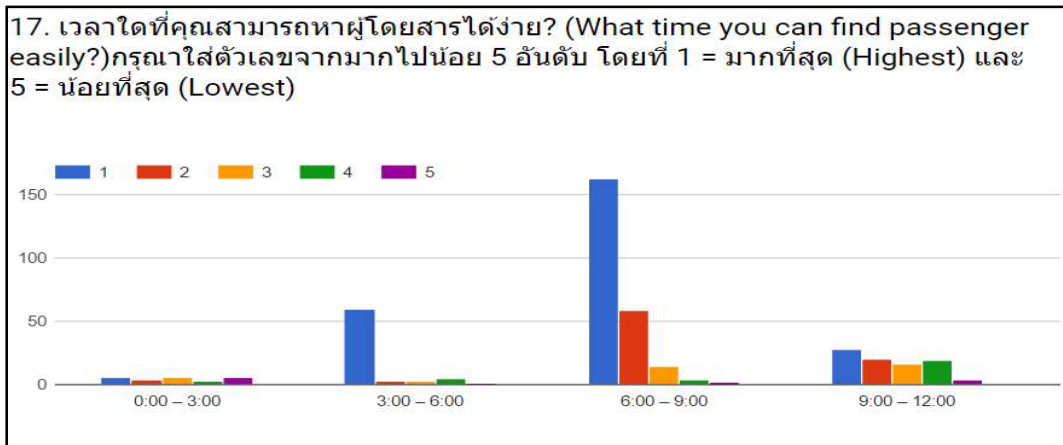
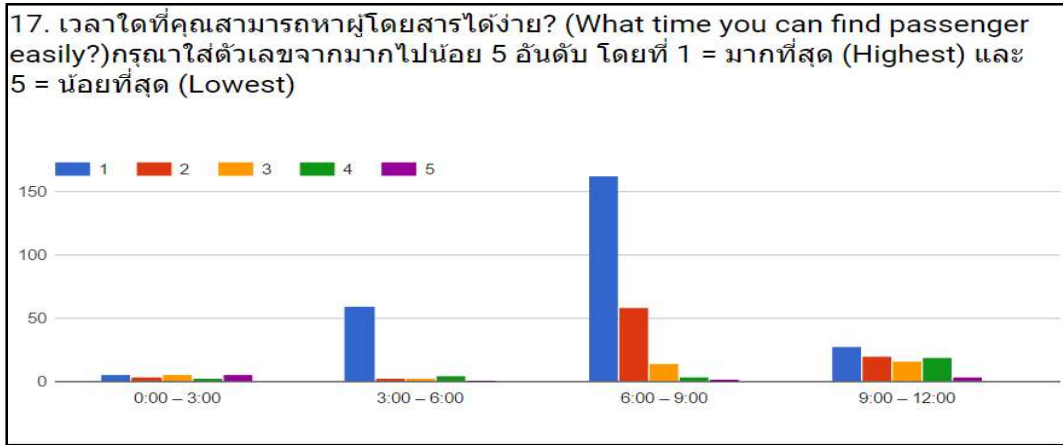


**Section B: Taxi Working Information**

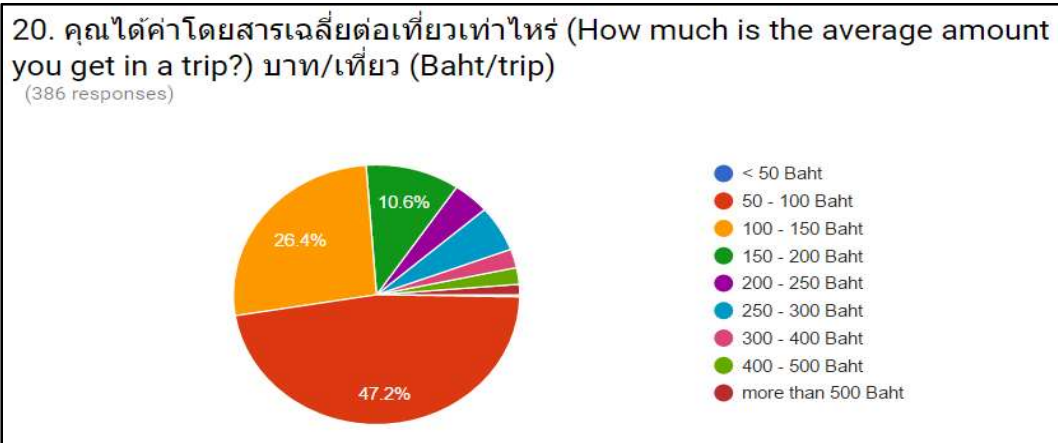




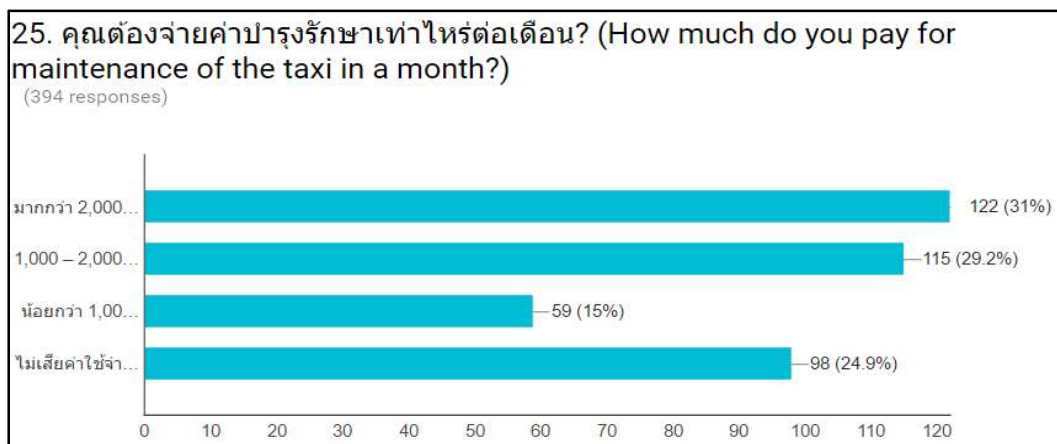
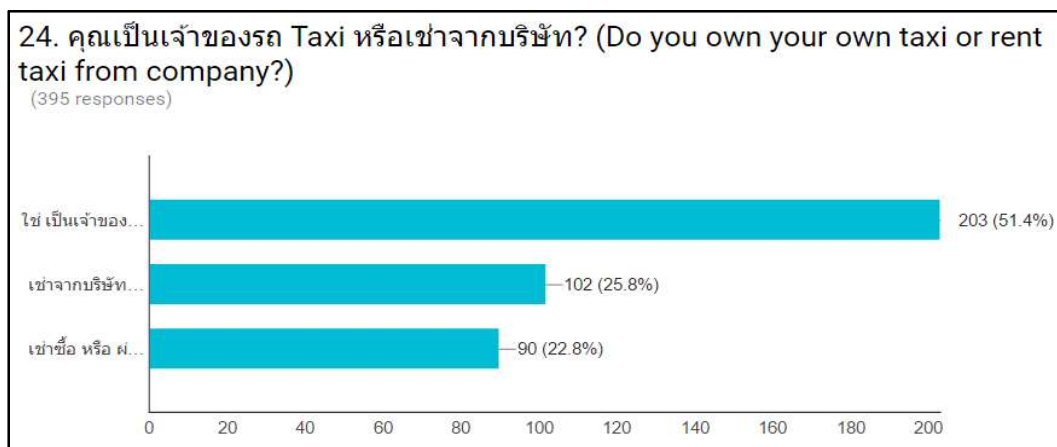


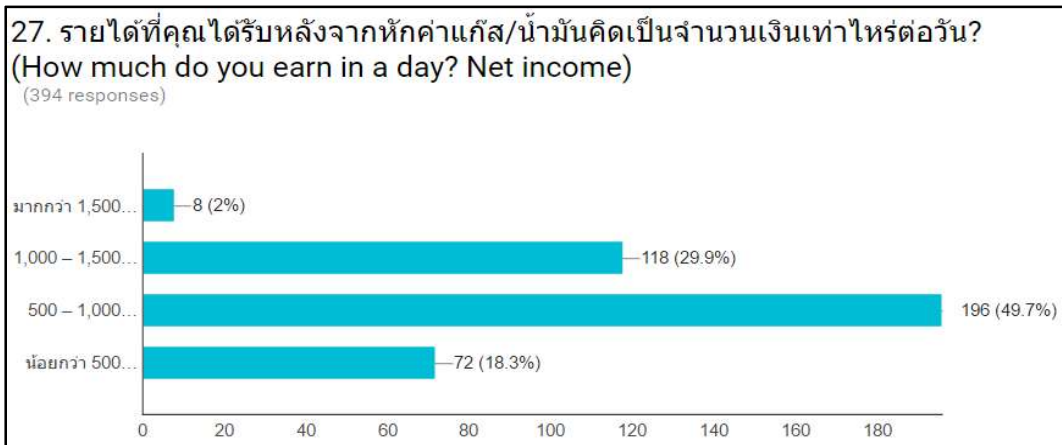
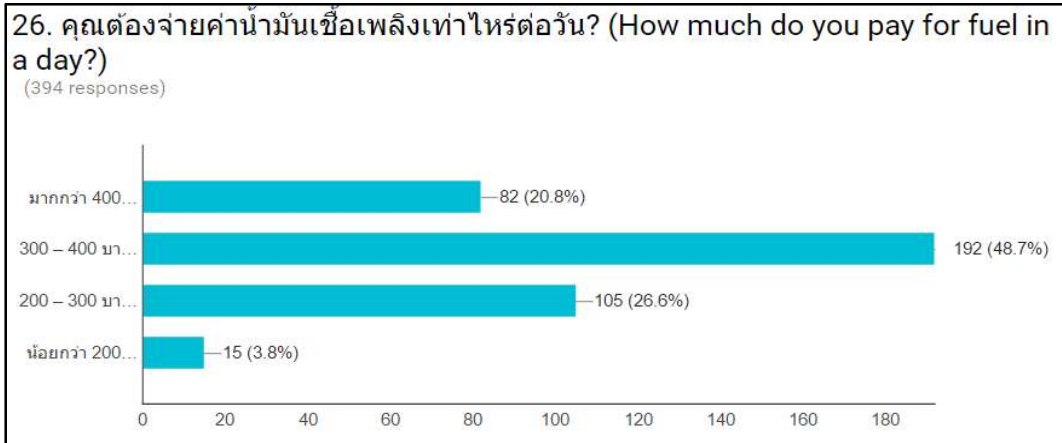




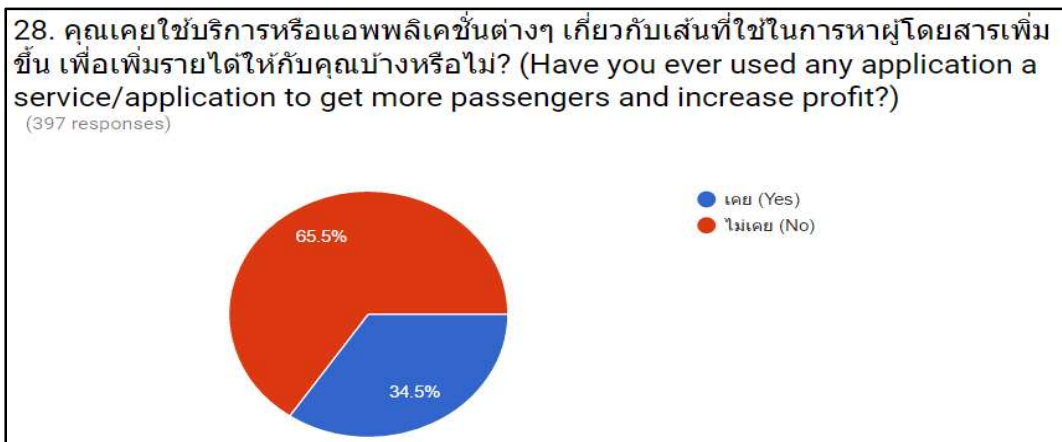


### Section C: Expenses



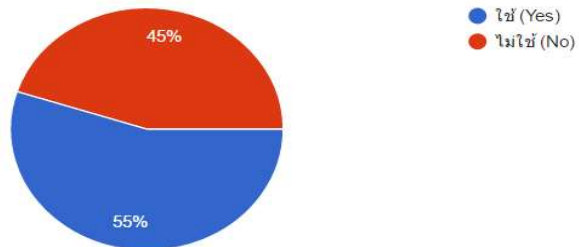


## Section D: Application



ถ้ามีบริการหรือแอปพลิเคชันตัวไหนที่คุณใช้ในการช่วยหาผู้โดยสารคุณจะใช้หรือไม่  
(Will you use application which we will develop that will optimize taxi route to get more passenger and increase profit?)

(389 responses)



## APPENDIX D

### Horton Data Platform

#### System Configuration

Operating System: CentOS v 7

Web Page: <https://www.centos.org/download/>

Hardware Raid: Setup the hard ware Raid 5 for Master Node prior to the installation of the Operating System. After Raid has been configured on the Master Node proceed with the installation of Centos 7 using the DVD ISO disk. Use Gnome Desktop for Master Node and Infrastructure Server for the Slave Nodes.

Network Configuration: Setup the network for all the host nodes and provide the IP address. For all the hosts nodes set up virtual IP address as well. The benefit of virtual IP address is that in the case the physical IP address gets changes; the system would still work based on virtual IP address configuration.

Format Hard Driver: Format all the available hard driver in all the hosts if hard driver is not formatted and mount the formatted hard drive.

#### Minimum System Requirement

The system requires browser capability as the for the web-based application. So, the operating system should have at least one web browser.

Software Requirement: Each host i.e. both Master and Slave nodes requires the following software installed

- yum
- rpm
- scp

- curl
- unzip
- tar
- wget
- python 2.7.x or higher
- Java 7 or higher

Database Requirement: Horton data platform for Ambari requires a relational database to store information about the clusters. Relational data base could be Hive, Oozie or PostgreSQL.

## **Environment Setup**

Before installing HDP through Ambari could, various environment needs to be set up.

Password Less SSH: For Ambari server to automatically install Ambari in all the hosts, password less SSH needs to be setup. SSH public key authentication is utilized for remote access during installation.

Network Time Protocol (NTP) Server: NTP server is required in all the hosts make clock of all the nodes synchronized to each other.

Fully Qualified Domain Name: For Ambari to communicate between the hosts, a fully qualified domain name need to be set up <fully.qualified.domain.name>.

Iptables Configuration: For Ambari to communicate between the hosts to deploy and manage posts during setup, Iptables need to be disable for seamless communication. If Iptables is running, then warning is displayed recommending user to disable it. In addition to the disabling the Iptables SELinux also needs to be disable in all the host in the cluster. Finally, umask in each of the hosts needs to be set as well accordingly.

## Ambari Server Installation

To setup HDP, Ambari needs to be installed in all the host of the cluster. Following is the step to setup the Ambari:

- Ambari repository download.
- Ambari server setup
- Ambari server start up

Download Ambari repository file to all the host as

```
'sudo wget -nv  
http://public-repo-1.hortonworks.com/ambari/centos7/2.x/updates/2.4.3.0/ambari.repo -O  
/etc/yum.repos.d/ambari.repo'  
'sudo yum install ambari-server'
```

### Ambari server setup

```
'sudo ambari-server setup'
```

### Ambari server start up

```
'sudo ambari-server start'
```

```
'sudo ambari-server status'
```

```
'sudo ambari-server stop'
```

## Ambari Agent Installation

To set up Ambari agent, download Ambari Repo in the host of the cluster and install the agent.

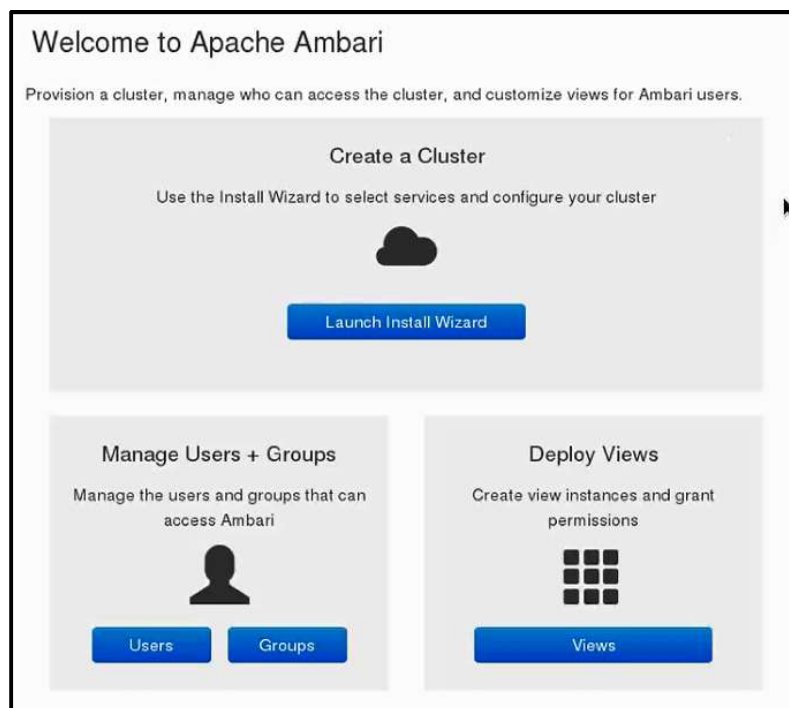
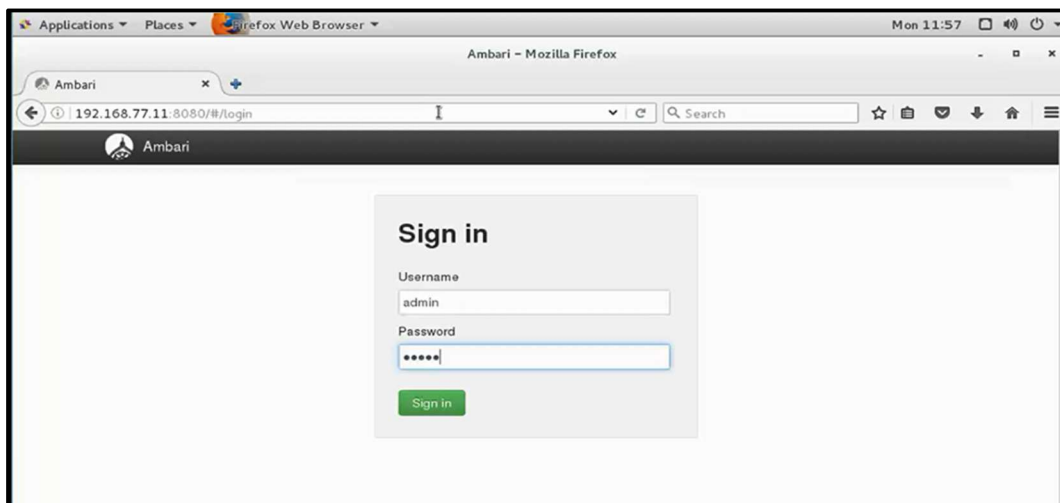
```
'sudo yum install ambari-agent'
```

Configure the Ambari Agent by editing the ambari-agent.ini to associate the ambari server IP address.

Ambari agent start in all host  
'sudo ambari-agent start'

## Installing, Configuring, and Deploying a HDP Cluster

1. Log in to Apache Ambari with default username admin from web browser



## 2. Name Your Cluster

CLUSTER INSTALL WIZARD

- Get Started
- Select Version
- Install Options
- Confirm Hosts
- Choose Services
- Assign Masters
- Assign Slaves and Clients
- Customize Services
- Review
- Install, Start and Test
- Summary

### Get Started

This wizard will walk you through the cluster installation process. First, start by naming your new cluster.

Name your cluster [Learn more](#)

Next >>

## 3. Select Version

### Select Version

Select the software version and method of delivery for your cluster. Using a Public Repository requires Internet connectivity. Using a Local Repository requires you have configured the software in a repository available in your network.

HDP-2.5

- HDP-2.4
- HDP-2.3
- HDP-2.2

Component	Version
Accumulo	1.7.0
Ambari Infra	0.1.0
Ambari Metrics	0.1.0
Atlas	0.7.0
Falcon	0.10.0
Flume	1.5.2
HBase	1.1.2

Use Public Repository  Use Local Repository

Use Public Repository  Use Local Repository

### Repositories

Provide Base URLs for the Operating Systems you are configuring.

OS	Name	Base URL	
redhat7	HDP-2.5	http://public-repo-1.hortonworks.com/HDP/centos7/2.x/upd:	+ Add - Remove
redhat7	HDP-UTILS-1.1.0.21	http://public-repo-1.hortonworks.com/HDP-UTILS-1.1.0.21/	

Skip Repository Base URL validation (Advanced) [?](#)

Use RedHat Satellite/Spacewalk [?](#)



## 4. Install Options

### Install Options

Enter the list of hosts to be included in the cluster and provide your SSH key.

**Target Hosts**

Enter a list of hosts using the Fully Qualified Domain Name (FQDN), one per line. Or use [Pattern Expressions](#)

```
ITU-hadoop1.ut.japan
ITU-hadoop2.ut.japan
ITU-hadoop3.ut.japan
```

**Host Registration Information**

Provide your **SSH Private Key** to automatically register hosts

No file selected.

ssh private key

SSH User Account:

SSH Port Number:

Perform [manual registration](#) on hosts and do not use SSH

## 5. Confirm Hosts

### Confirm Hosts

Registering your hosts.  
Please confirm the host list and remove any hosts that you do not want to include in the cluster.

Show: **All (3)** | [Installing \(0\)](#) | [Registering \(3\)](#) | [Success \(0\)](#) | [Fail \(0\)](#)

<input type="checkbox"/> Host	Progress	Status	Action
<input type="checkbox"/> itu-hadoop1.ut.japan	<div style="width: 100%;"></div>	Registering	<input type="button" value="Remove"/>
<input type="checkbox"/> itu-hadoop2.ut.japan	<div style="width: 100%;"></div>	Registering	<input type="button" value="Remove"/>
<input type="checkbox"/> itu-hadoop3.ut.japan	<div style="width: 100%;"></div>	Registering	<input type="button" value="Remove"/>

Show: 25 | 1 - 3 of 3 |

## 6. Choose Services

### Choose Services

Choose which services you want to install on your cluster.

<input type="checkbox"/> Service	Version	Description
<input checked="" type="checkbox"/> HDFS	2.7.3	Apache Hadoop Distributed File System
<input checked="" type="checkbox"/> YARN + MapReduce2	2.7.3	Apache Hadoop NextGen MapReduce (YARN)
<input checked="" type="checkbox"/> Tez	0.7.0	Tez is the next generation Hadoop Query Processing framework written on top of YARN.
<input checked="" type="checkbox"/> Hive	1.2.1000	Data warehouse system for ad-hoc queries & analysis of large datasets and table & storage management service
<input checked="" type="checkbox"/> HBase	1.1.2	A Non-relational distributed database, plus Phoenix, a high performance SQL layer for low latency applications.
<input checked="" type="checkbox"/> Pig	0.16.0	Scripting platform for analyzing large datasets
<input checked="" type="checkbox"/> Sqoop	1.4.6	Tool for transferring bulk data between Apache Hadoop and structured data stores such as relational databases
<input checked="" type="checkbox"/> Oozie	4.2.0	System for workflow coordination and execution of Apache Hadoop jobs. This also includes the installation of the optional Oozie Web Console which relies on and will install the <a href="#">ExtJS</a> Library.
<input checked="" type="checkbox"/> ZooKeeper	3.4.6	Centralized service which provides highly reliable distributed coordination
<input checked="" type="checkbox"/> Falcon	0.10.0	Data management and processing platform
<input checked="" type="checkbox"/> Storm	1.0.1	Apache Hadoop Stream processing framework

## 7. Assign Masters

### Assign Masters

Assign master components to hosts you want to run them on.  
+ HiveServer2 and WebHCat Server will be hosted on the same host.

SNameNode: itu-hadoop2.ut.japan (3.7 GB, 2 c)

NameNode: itu-hadoop1.ut.japan (4.5 GB, 2 c)

App Timeline Server: itu-hadoop2.ut.japan (3.7 GB, 2 c)

ResourceManager: itu-hadoop2.ut.japan (3.7 GB, 2 c)

History Server: itu-hadoop2.ut.japan (3.7 GB, 2 c)

Hive Metastore: itu-hadoop2.ut.japan (3.7 GB, 2 c)

WebHCat Server: itu-hadoop2.ut.japan+

itu-hadoop1.ut.japan (4.5 GB, 2 cores)


- NameNode
- HBase Master
- ZooKeeper Server
- Infra Solr Instance
- Grafana
- Activity Analyzer
- Activity Explorer
- HST Server
- Spark History Server

itu-hadoop2.ut.japan (3.7 GB, 2 cores)

- SNameNode
- App Timeline Server
- ResourceManager
- History Server
- Hive Metastore
- WebHCat Server
- HiveServer2
- ZooKeeper Server

## 8. Assign Slaves and Clients

### Assign Slaves and Clients

Assign slave and client components to hosts you want to run them on.  
Hosts that are assigned master components are shown with .  
"Client" will install HDFS Client, YARN Client, MapReduce2 Client, Tez Client, HCat Client, Hive Client, HBase Client, Pig Client, ZooKeeper Client, Infra Solr Client, Spark Client and Slider Client.

	none	all	none	all	none	all	none	all	none	all	none	all	none
NFSGateway	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
NodeManager	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
RegionServer	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
Phoenix Query Server	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
Livy Server	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
Spark Thrift Server	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
Client	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>

Show: 25 1 of 3

[- Back](#) [Next ->](#)

## 9. Customize Services

### Customize Services

We have come up with recommended configurations for the services you selected. Customize them as you see fit.

HDFS YARN MapReduce2 Tez Hive **1** HBase Pig ZooKeeper Ambari Infra Ambari Metrics **1**  
SmartSense **1** Spark Slider Misc

Group: Default (3) [Manage Config Groups](#) Filter...

Settings [Advanced](#)

#### NameNode

NameNode directories

NameNode Java heap size

0GB 1GB 2.25GB 4.452GB

#### DataNode

DataNode directories

DataNode failed disk tolerance

0 1

## 10. Review

### Review

Please review the configuration before installation

**Admin Name :** hadoop  
**Cluster Name :** ITU\_Hadoop  
**Total Hosts :** 3 (3 new)

**Repositories:**

- redhat7 (HDP-2.5):  
<http://public-repo-1.hortonworks.com/HDP/centos7/2.x/updates/2.5.3.0>
- redhat7 (HDP-UTILS-1.1.0.21):  
<http://public-repo-1.hortonworks.com/HDP-UTILS-1.1.0.21/repos/centos7>

**Services:**

**HDFS**

- DataNode : 2 hosts
- NameNode : itu-hadoop1.ut.japan
- NFSGateway : 0 host
- SNameNode : itu-hadoop2.ut.japan

**YARN + MapReduce2**

- App Timeline Server : itu-hadoop1.ut.japan
- NodeManager : 2 hosts
- ResourceManager : itu-hadoop1.ut.japan

**Tez**

[-- Back](#) [Print](#) [Deploy --](#)

## 11. Install, Start and Test

### Install, Start and Test

Please wait while the selected services are installed and started.

4% overall

Show: [All \(3\)](#) | [In Progress \(3\)](#) | [Warning \(0\)](#) | [Success \(0\)](#) | [Fail \(0\)](#)

Host	Status	Message
itu-hadoop1.ut.japan	<div style="width: 4%;"><div style="width: 4%;"></div></div> 4%	Installing Activity Analyzer
itu-hadoop2.ut.japan	<div style="width: 5%;"><div style="width: 5%;"></div></div> 5%	Installing DataNode
itu-hadoop3.ut.japan	<div style="width: 4%;"><div style="width: 4%;"></div></div> 4%	Installing DataNode

3 of 3 hosts showing - [Show All](#) Show: 25 1 - 3 of 3 [←](#) [→](#)

[Next --](#)

## Install, Start and Test

Please wait while the selected services are installed and started.

100 % overall

Show: All (3) | In Progress (0) | Warnings (2) | Success (1) | Fail (0)

Host	Status	Message
itu-hadoop1.ut.japan	100%	Warnings encountered
itu-hadoop2.ut.japan	100%	Success
itu-hadoop3.ut.japan	100%	Warnings encountered

3 of 3 hosts showing - [Show All](#) Show: 25 1 - 3 of 3

Installed and started the services with some warnings.

[Next](#)

12. Complete

## Summary

Here is the summary of the install process.

The cluster consists of 3 hosts  
 Installed and started services successfully on 1 new host  
 2 warnings

Master services installed

- SNameNode installed on itu-hadoop2.ut.japan
- NameNode installed on itu-hadoop1.ut.japan
- ResourceManager installed on itu-hadoop1.ut.japan
- History Server installed on itu-hadoop1.ut.japan
- HiveServer2 installed on itu-hadoop1.ut.japan
- HBase Master installed on itu-hadoop1.ut.japan

Starting services failed

[Complete](#)

## APPENDIX E

### HADOOP-HIVE QUERY FOR TRIP ORIGIN DESTINATION

#### Taxi Trip Origin and Destination

**-- create origin destination matrix based on grid data for each imei and partition by dateonly in array data type**

```
drop table taxigps_organ_destination;
```

```
create table taxigps_organ_destination (imei string, record array<array<string>>)
partitioned by (dateonly string)
row format delimited
fields terminated by '\t'
collection items terminated by ','
map keys terminated by '!'
lines terminated by '\n'
stored as sequencefile;
```

```
insert overwrite table taxigps_organ_destination partition (dateonly='2015-06-01')
select imei, origindestinationmatrix (record, '/grid_data_location') as record from
taxigps_filter_duplicate_data where record is not null and size(record) > 0 and dateonly = '2015-
06-01';
```

**-- convert origin destination array data type to lateral view data type**

```
drop table taxigps_organ_destination_lateral;
```

```
create table taxigps_organ_destination_lateral (get_in_grid string, get_off_grid string, od_count
int)
row format delimited
```

```
fields terminated by '\t'  
collection items terminated by ','  
map keys terminated by '!'  
lines terminated by '\n'  
stored as sequencefile;
```

```
insert overwrite table taxigps_organ_destination_lateral  
select split(concat_ws(',', value),",")[0], split(concat_ws(',', value),",")[1], split(concat_ws(',',  
value),",")[2] from taxigps_organ_destination lateral view explode(record) adtable as value;
```

```
-- compute the overall aggregated origin destination count from one grid id to another grid  
id
```

```
drop table taxigps_organ_destination_aggregated;
```

```
create table taxigps_organ_destination_aggregated (get_in_grid string, get_off_grid string,  
od_count int)  
row format delimited  
fields terminated by '\t'  
collection items terminated by ','  
map keys terminated by '!'  
lines terminated by '\n'  
stored as sequencefile;
```

```
insert overwrite table taxigps_organ_destination_aggregated  
select get_in_grid, get_off_grid, sum(od_count) from taxigps_organ_destination_lateral group by  
get_in_grid, get_off_grid;
```

**-- copy the table data from hive to local directory**

```
insert overwrite local directory 'location_origin_destination'  
row format delimited  
fields terminated by ','  
collection items terminated by '  
select * from taxigps_organ_destination_aggregated where od_count>1;
```

**-- create origin destination matrix for a trip based on grid data for each imei and partition by dateonly in array data type and time range. od with passenger trip and no passenger trip information**

```
drop table taxigps_organ_destination_trip;
```

```
create table taxigps_organ_destination_trip (imei string, record array<array<string>>)  
partitioned by (dateonly string, daytype string, weathertype string, eventtype string)  
row format delimited  
fields terminated by '\t'  
collection items terminated by ','  
map keys terminated by '!'  
lines terminated by '\n'  
stored as sequencefile;
```

```
insert overwrite table taxigps_organ_destination_trip partition (dateonly='2015-06-01',  
daytype='weekdays', weathertype='n', eventtype='n') select imei, origindestinationmatrixtrip  
(record,'location_grid_data) as record from taxigps_filter_duplicate_data where record is not null  
and size(record) > 0 and dateonly = '2015-06-01';
```



**-- convert orgin destination trip array data type to lateral view data type partition by dateonly, daytype, weathertype and eventtype of each imei**

```
drop table taxigps_orgin_destination_trip_lateral;
```

```
create table taxigps_orgin_destination_trip_lateral (imei string, get_in_grid string, get_off_grid string, trip_start_time string, trip_end_time string, trip_speed string, trip_time string, trip_type string) partitioned by (dateonly string, daytype string, weathertype string, eventtype string) row format delimited fields terminated by '\t' collection items terminated by ',' map keys terminated by '!' lines terminated by '\n' stored as sequencefile;
```

**-- query without daytype, weathertype and eventtype**

```
insert overwrite table taxigps_orgin_destination_trip_lateral select split(concat_ws(',', value),",")[0], split(concat_ws(',', value),",")[1], split(concat_ws(',', value),",")[2] , split(concat_ws(',', value),",")[3], split(concat_ws(',', value),",")[4], split(concat_ws(',', value),",")[5], split(concat_ws(',', value),",")[6], split(concat_ws(',', value),",")[7] from taxigps_orgin_destination_trip_lateral view explode(record) adtable as value where size(record) > 0;
```

**-- aggregate orgin and destination based on get in grid and time with trip type as partition**

```
drop table taxigps_orgin_destination_trip_aggregated;
```

```
create table taxigps_orgin_destination_trip_aggregated (timerange string, get_in_grid string, get_off_grid string, od_count int)
```

partitioned by (trip\_type string, daytype string)  
row format delimited  
fields terminated by '\t'  
collection items terminated by ','  
map keys terminated by '!'  
lines terminated by '\n'  
stored as sequencefile;

```
insert overwrite table taxigps_organ_destination_trip_aggregated partition (trip_type =  
'passengertrip', daytype='weekdays')  
select hour(trip_start_time) as timerange, get_in_grid, get_off_grid, count(imei) from  
taxigps_organ_destination_trip_lateral where trip_type = 'passengertrip' and daytype='weekdays'  
group by hour(trip_start_time), get_in_grid, get_off_grid order by get_in_grid;
```

**-- aggregate each organ grid based on get in grid and time**

```
drop table taxigps_organ_trip_aggregated;
```

```
create table taxigps_organ_trip_aggregated (timerange string, get_in_grid string, o_count int)  
partitioned by (trip_type string, daytype string)  
row format delimited  
fields terminated by '\t'  
collection items terminated by ','  
map keys terminated by '!'  
lines terminated by '\n'  
stored as sequencefile;
```

```
insert overwrite table taxigps_organ_trip_aggregated partition (trip_type = 'passengertrip', daytype  
= 'weekdays')  
select timerange, get_in_grid, sum(od_count) from taxigps_organ_destination_trip_aggregated  
where trip_type = 'passengertrip' and daytype='weekdays' group by timerange, get_in_grid;
```

**-- transition probability matrix table based on origin destination for each grid transition to next grid at each time step for both trip and notrip**

```
create table taxigps_origin_destination_transition (timerange string, get_in_grid string,
get_off_grid string, od_count int, o_count int, od_probability double)
partitioned by (trip_type string, daytype string)
row format delimited
fields terminated by '\t'
collection items terminated by ','
map keys terminated by '!'
lines terminated by '\n'
stored as sequencefile;
```

```
insert overwrite table taxigps_origin_destination_transition partition (trip_type='passengertrip',
daytype = 'weekdays')
select a.timerange, a.get_in_grid, a.get_off_grid, a.od_count, b.o_count, a.od_count/b.o_count
from
taxigps_origin_destination_trip_aggregated a,
taxigps_origin_trip_aggregated b
where a.get_in_grid = b.get_in_grid and
a.timerange = b.timerange and
a.trip_type = b.trip_type and
a.daytype = b.daytype and
a.trip_type = 'passengertrip' and
a.daytype = 'weekdays';
```

**APPENDIX F**

**EXTRATED SECONDARY DATA FROM GPS PROBE**

**Probe GPS Data Map Matched Sample**

<b>IMEI</b>	<b>Original Latitude</b>	<b>Original Longitude</b>	<b>Timestamp</b>	<b>Map Matched Link Id</b>	<b>Map Matched Latitude</b>	<b>Map Matched Longitude</b>
353419036175292	13.79268	100.7218	5:36:07	22912	13.792859	100.722264
353419036175292	13.79274	100.72178	5:36:10	22912	13.792918	100.722241
353419036175292	13.79296	100.72169	5:36:19	22912	13.79314	100.722156
353419036175292	13.79335	100.72156	5:36:37	22912	13.793523	100.722008
353419036175292	13.7934	100.72155	5:36:39	22912	13.79357	100.721989
353419036175292	13.79344	100.72153	5:36:42	22912	13.793611	100.721973
353419036175292	13.7935	100.7215	5:36:45	22912	13.793674	100.721949
353419036175292	13.79356	100.72147	5:36:48	22912	13.793736	100.721925
353419036175292	13.79362	100.72144	5:36:51	22912	13.793798	100.721901
353419036175292	13.79479	100.72105	5:37:43	22912	13.794947	100.721458
353419036175292	13.79482	100.72115	5:37:46	22912	13.79494	100.72146
353419036175292	13.79484	100.72122	5:37:49	22912	13.794934	100.721463
353419036175292	13.79488	100.72131	5:37:52	22912	13.794938	100.721461
353419036175292	13.79491	100.72138	5:37:55	22912	13.794941	100.72146
353419036175292	13.79495	100.72141	5:37:58	22912	13.794966	100.72145
353419036175292	13.79504	100.72141	5:38:01	22912	13.795044	100.72142
353419036175292	13.79517	100.72136	5:38:04	22912	13.795174	100.72137
353419036175292	13.79531	100.72131	5:38:07	22912	13.795313	100.721317
353419036175292	13.79546	100.72125	5:38:10	22912	13.795463	100.721258
353419036175292	13.79557	100.72121	5:38:13	22912	13.795572	100.721216

## Taxi Agent Data Sample

Agent IMEI	Grid Id	Latitude	Longitude	Grid Geometry	Start Time
10000023	200000039739	13.709517	100.36367	Polygon ((100.359604 13.710102, 100.364226 13.710102, 100.364226 13.705581, 100.359604 13.705581, 100.359604 13.710102))	0:00:00
10008773	200000042094	13.680176	100.495436	Polygon ((100.493642 13.682976, 100.498264 13.682976, 100.498264 13.678455, 100.493642 13.678455, 100.493642 13.682976))	0:27:35
10008833	200000042864	13.673565	100.491946	Polygon ((100.489020 13.673934, 100.493642 13.673934, 100.493642 13.669413, 100.489020 13.669413, 100.489020 13.673934))	0:00:00
10008908	200000036693	13.741863	100.605691	Polygon ((100.604570 13.746270, 100.609192 13.746270, 100.609192 13.741749, 100.604570 13.741749, 100.604570 13.746270))	0:00:01
10008917	200000031418	13.811317	100.651569	Polygon ((100.650790 13.814085, 100.655412 13.814085, 100.655412 13.809564, 100.650790 13.809564, 100.650790 13.814085))	13:23:51
10008930	200000048800	13.602851	100.879059	Polygon ((100.877268 13.606119, 100.881890 13.606119, 100.881890 13.601598, 100.877268 13.601598, 100.877268 13.606119))	16:29:51

### Direction Probability Data Sample

Grid Id	Time Interval	Direction	Direction Probability	Day Type
200000001392	270	South	1	Weekdays
200000001416	975	West	1	Weekdays
200000001426	1190	West	0.923	Weekdays
200000001484	190	North	0.625	Weekdays
200000001530	1200	North West	0.6	Weekdays
200000001564	545	South	0.25	Weekdays
200000001617	750	West	1	Weekdays
200000001676	895	South East	1	Weekdays
200000001693	830	South	0.606	Weekdays
200000001751	1405	North	1	Weekdays
200000001813	680	North East	0.615	Weekdays
200000001860	1260	East	0.5	Weekdays
200000001877	670	South	0.5	Weekdays
200000002003	880	North	0.706	Weekdays
200000002127	795	North West	0.389	Weekdays
200000002248	215	North West	1	Weekdays
200000002457	855	North	1	Weekdays
200000002544	930	South	0.778	Weekdays
200000002626	635	South West	0.5	Weekdays
200000002648	415	South East	0.667	Weekdays
200000002663	1000	East	0.667	Weekdays
200000002668	845	South	1	Weekdays
200000002747	55	North	0.75	Weekdays
200000002847	840	South East	0.889	Weekdays
200000002917	980	South West	1	Weekdays

### Demand Probability of Success Data Sample

Total Pick Up	Total Vacant Taxi	Probability of Success	Time Interval	Grid Id	Day Type
0	4	0	0	200000047513	Weekends
0	6	0	0	200000025550	Weekdays
0	1	0	0	200000019542	Weekdays
1	50	0.02	0	200000032424	Weekends
0	8	0	0	200000015905	Weekdays
0	6	0	0	200000062241	Weekdays
0	8	0	0	200000017014	Weekdays
5	140	0.0357143	0	200000038672	Weekends
16	171	0.0935673	0	200000047155	Weekends
0	1	0	0	200000018127	Weekdays
0	9	0	0	200000022649	Weekdays
4	127	0.0314961	0	200000033106	Weekdays
6	75	0.08	0	200000035901	Weekdays
32	640	0.05	0	200000029131	Weekdays
1	38	0.0263158	0	200000049470	Weekdays
1	48	0.0208333	0	200000054472	Weekends
0	43	0	0	200000027849	Weekends
0	3	0	0	200000025308	Weekends
0	7	0	0	200000035895	Weekdays
0	22	0	0	200000025322	Weekends
0	1	0	0	200000046735	Weekdays
389	877	0.444	0	200000037856	Weekdays
11	73	0.151	0	200000033424	Weekends
8	52	0.154	0	200000042870	Weekends
90	276	0.326	0	200000031395	Weekends

**Origin Destination Probability Data Sample**

<b>Time Interval for Every One Hour</b>	<b>Origin Grid</b>	<b>Destination Grid</b>	<b>Total Trip Between Origin Grid to Destination</b>	<b>Total Trip Originated in Origin Grid</b>	<b>Origin Destination Probability</b>	<b>Trip Type</b>	<b>Day Type</b>
4	200000020295	200000009391	24	52	0.462	Passenger Trip	Week days
4	200000020297	200000024229	41	526	0.078	Passenger Trip	Week days
5	200000016332	200000010275	29	31	0.935	Passenger Trip	Week days
5	200000019599	200000019599	31	44	0.705	Passenger Trip	Week days
6	200000020297	200000024229	23	322	0.071	Passenger Trip	Week days
12	200000020297	200000024229	23	459	0.050	Passenger Trip	Week days
14	200000017721	200000020297	22	99	0.222	Passenger Trip	Week days
14	200000020297	200000024229	36	526	0.068	Passenger Trip	Week days
16	200000017721	200000020297	34	145	0.234	Passenger Trip	Week days
16	200000020297	200000024229	22	652	0.034	Passenger Trip	Week days
9	200000036679	200000040994	21	102	0.206	Passenger Trip	Week ends



## Trip Data Sample

<b>IMEI</b>	<b>Pick Up Latitude</b>	<b>Pick Up Longitude</b>	<b>Pick Up Time</b>	<b>Drop Off Latitude</b>	<b>Drop Off Longitude</b>	<b>Drop Off Time</b>
10011935	13.73412	100.56743	0:12:10	13.71458	100.59465	0:24:01
10011935	13.73393	100.57975	0:36:10	13.72167	100.5792	0:44:05
10011935	13.73665	100.57443	0:54:56	13.7377	100.5834	0:58:08
10011935	13.72415	100.5793	1:12:44	13.75008	100.53855	1:30:30
10011935	13.73818	100.5587	1:50:07	13.7058	100.60128	2:04:26
10011935	13.70505	100.6014	2:05:37	13.69995	100.67555	2:23:12
10011935	13.73843	100.55853	3:00:25	13.70968	100.62583	3:18:42
10011935	13.73427	100.58753	4:06:48	13.82668	100.65183	4:38:28
10011935	13.7698	100.62268	5:08:27	13.72802	100.54048	5:31:06
10011935	13.71995	100.56013	5:35:49	13.67495	100.6631	5:53:05
10011935	13.69373	100.63912	6:14:10	13.7064	100.60075	6:23:00
10011935	13.71218	100.60573	6:31:25	13.70772	100.61438	6:34:27
10011935	13.72905	100.59615	7:04:55	13.73325	100.56722	7:15:34
10011935	13.73878	100.5925	7:49:19	13.71932	100.5612	8:04:03
10011935	13.71968	100.56033	8:05:26	13.80522	100.56805	8:26:39
10011935	13.81208	100.563	8:32:48	13.8043	100.56733	8:37:56
10011935	13.80218	100.58078	8:51:44	13.8053	100.58333	8:54:56
10011935	13.822	100.57937	9:01:59	13.77823	100.56807	9:11:47
10011935	13.77392	100.56997	9:14:33	13.78392	100.57723	9:19:57
10011935	13.7713	100.5836	9:32:12	13.92012	100.60193	10:16:48
10011935	13.87127	100.6028	10:49:43	13.86855	100.59242	10:59:44
10011935	13.88162	100.58602	11:09:42	13.8034	100.55423	11:21:09
10011935	13.79073	100.54352	11:38:04	13.7467	100.52805	12:02:44
10011935	13.74602	100.5527	12:20:17	13.73257	100.5639	12:32:04
10011935	13.7323	100.56385	12:34:30	13.741	100.56085	12:41:50
10011935	13.72772	100.58545	13:34:28	13.72632	100.57655	13:42:38

### Grid Speed Data Sample

Grid ID	Time Interval Fifteen Minutes	Grid Speed	Day Type
200000001058	270	98.5	Weekdays
200000001058	435	89.33	Weekdays
200000001058	465	86	Weekdays
200000001058	585	77	Weekdays
200000001058	600	102.75	Weekdays
200000001058	615	99.6	Weekdays
200000001058	675	75	Weekdays
200000001058	720	75.14	Weekdays
200000001058	735	98.5	Weekdays
200000001058	810	89	Weekdays
200000001058	825	89.5	Weekdays
200000001058	840	71.69	Weekdays
200000001058	915	83.75	Weekdays
200000001058	930	75.64	Weekdays
200000001058	945	84.7	Weekdays
200000001058	960	70.67	Weekdays
200000001058	975	84	Weekdays
200000001058	1020	80	Weekdays
200000001058	1050	88.5	Weekdays
200000001058	1065	70	Weekdays
200000001058	1080	74.4	Weekdays
200000001058	1125	89	Weekdays
200000001058	1140	73.2	Weekdays
200000001058	1155	61.5	Weekdays
200000001058	1275	97	Weekdays
200000001059	270	95.5	Weekdays

### OSM Link Speed Data Sample

OSM Link ID	Time Interval Fifteen Minutes	Link Speed	Day Type
108256	225	29.25	Weekdays
108256	240	47	Weekdays
108256	255	44.42	Weekdays
108256	270	45.44	Weekdays
108256	285	36.17	Weekdays
108256	300	36.73	Weekdays
108256	315	44	Weekdays
108256	330	35.25	Weekdays
108256	345	34.62	Weekdays
108256	360	37.33	Weekdays
108256	375	34.86	Weekdays
108256	390	29.29	Weekdays
108256	405	30.65	Weekdays
108256	420	31.79	Weekdays
108256	435	27.11	Weekdays
108256	450	23.51	Weekdays
108256	465	25.28	Weekdays
108256	480	21.8	Weekdays
108256	495	23.96	Weekdays
108256	510	23.59	Weekdays
108256	525	28.36	Weekdays
108256	540	22.75	Weekdays
108256	555	24.08	Weekdays
108256	570	25.86	Weekdays
108256	585	25.7	Weekdays
108256	600	23.97	Weekdays





## REFERENCES

- Abar, S., Theodoropoulos, G. K., Lemarini, P., & O'Hare, G. M. P. (2017). Agent Based Modelling and Simulation tools: A review of the state-of-art software. *Computer Science Review, 24*, 13–33. doi:10.1016/j.cosrev.2017.03.001
- Apache Hadoop YARN. (n.d.). <https://hortonworks.com/apache/yarn/>
- Apache Hive. (2013). <https://hive.apache.org/index.html>
- Balan, R. K., Woodard, C. J., & Santani, D. (2008). Understanding and Improving a GPS-based Taxi System. *The Sixth International Conference on Mobile Systems, Applications and Services, Breckenridge, Colorado*. [http://apollo.smu.edu.sg/papers/MobiSys08\\_poster.pdf](http://apollo.smu.edu.sg/papers/MobiSys08_poster.pdf)
- Baster, B., Duda, J., Maciol, A., & Rebiasz, B. (2013). Rule-based approach to human-like decision simulating in agent-based modeling and simulation. In *17th International Conference on System Theory, Control and Computing, ICSTCC 2013; Joint Conference of SINTES 2013, SACCS 2013, SIMSIS 2013 - Proceedings* (pp. 739–743). doi:10.1109/ICSTCC.2013.6689049
- Bharathy, G. K., & Silverman, B. (2012). *HOLISTICALLY EVALUATING AGENT BASED SOCIAL SYSTEMS MODELS: A Case Study. Most* (Vol. 89).
- Bischoff, J., Maciejewski, M., & Sohr, A. (2015). Analysis of Berlin's taxi services by exploring GPS traces. *International Conference on Models and Technologies for Intelligent Transportation Systems, MT-ITS 2015*, (June), 209–215. doi:10.1109/MTITS.2015.7223258
- Bonabeau, E. (2002). Agent-based modeling: methods and techniques for simulating human systems. *Proceedings of the National Academy of Sciences, 99*(suppl. 3), 7280–7287. doi:10.1073/pnas.082080899
- Brakatsoulas, S., Salas, R., & Wenk, C. (2005). On Map-Matching Vehicle Tracking Data. *Computing, 853–864*. doi:10.1109/CVPR.2009.5206848
- Campello, R. J. G. B., Moulavi, D., & Sander, J. (2013). Density-Based Clustering Based on Hierarchical Density Estimates. *Advances in Knowledge Discovery and Data Mining*, 160–172. doi:10.1007/978-3-642-37456-2\_14
- Castro, P. S., Zhang, D., Chen, C., Li, S., & Pan, G. (2013). From Taxi GPS Traces to Social and Community Dynamics: A Survey. *ACM Computing Surveys, 46*(2), 1–34. doi:10.1145/2543581.2543584

- Castro, P. S., Zhang, D., & Li, S. (2012). Urban Traffic Modelling and Prediction Using Large Scale Taxi GPS Traces. In J. Kay, P. Lukowicz, H. Tokuda, P. Olivier, & A. Krüger (Eds.), *Pervasive Computing* (pp. 57–72). Berlin, Heidelberg: Springer Berlin Heidelberg. doi:10.1007/978-3-642-31205-2\_4
- Chakka, V. P., Everspaugh, A. C., & Patel, J. M. (2003). Indexing Large Trajectory Data Sets With SETI. *CIDR Conference on Innovative Data Systems Research*. doi:10.1.1.12.182
- Chen, F., Shen, M., & Tang, Y. (2011). Local path searching based map matching algorithm for floating car data. *Procedia Environmental Sciences*, 10(PART A), 576–582. doi:10.1016/j.proenv.2011.09.093
- Cheng, S. F., & Nguyen, T. D. (2011). TaxiSim: A Multiagent Simulation Platform for Evaluating Taxi Fleet Operations. In *2011 IEEE/WIC/ACM International Conferences on Web Intelligence and Intelligent Agent Technology* (Vol. 2, pp. 14–21). doi:10.1109/WI-IAT.2011.138
- Cheng, Y. (1995). Mean Shift, Mode Seeking, and Clustering. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 17(8), 790–799. doi:10.1109/34.400568
- Comanicu, D., & Meer, P. (1999). Mean shift analysis and applications. *Proceedings of the Seventh IEEE International Conference on Computer Vision*, (2), 1197–1203 vol.2. doi:10.1109/ICCV.1999.790416
- De Mauro, A., Greco, M., & Grimaldi, M. (2014). What is Big Data? A Consensual Definition and a Review of Key Research Topics. In *4th International Conference on Integrated Information* (Vol. 1644, pp. 97–104). doi:10.13140/2.1.2341.5048
- Deng, Z., & Ji, M. (2011). Spatiotemporal structure of taxi services in Shanghai: Using exploratory spatial data analysis. In *19th International Conference on Geoinformatics* (pp. 1–5). doi:10.1109/GeoInformatics.2011.5981129
- Ester, M., Kriegel, H., Sander, J., & Xu, X. (1996). A density-based algorithm for discovering clusters a density-based algorithm for discovering clusters in large spatial databases with noise. In *Proceedings of 2nd International Conference on Knowledge Discovery and Data Mining (KDD-96)*. <http://citeseerx.ist.psu.edu/viewdoc/download?jsessionid=77A7B1B3F82E85401DB7CE60085A6D3F?doi=10.1.1.121.9220&rep=rep1&type=pdf>
- Francis, D. H., Madria, S., & Sabharwal, C. (2008). A scalable constraint-based Q-hash indexing

- for moving objects. *Information Sciences*, 178(6), 1442–1460. doi:10.1016/j.ins.2007.11.004
- Gan, J., & Tao, Y. (2015). DBSCAN Revisited: Mis-Claim, Un-Fixability, and Approximation. *Proceedings of the 2015 ACM SIGMOD International Conference on Management of Data*, 519–530. doi:10.1145/2723372.2737792
- Ge, Q., & Fukuda, D. (2016). Updating origin-destination matrices with aggregated data of GPS traces. *Transportation Research Part C: Emerging Technologies*, 69, 291–312. doi:10.1016/j.trc.2016.06.002
- geofabrik.de. (n.d.). <http://download.geofabrik.de/asia/thailand.html>. Accessed 4 June 2016
- Ghemawat, S., Gobioff, H., & Leung, S.-T. (2003). The Google file system. *Proceedings of the nineteenth ACM symposium on Operating systems principles - SOSP '03*, 29. doi:10.1145/945449.945450
- Goh, C., Dauwels, J., & Mitrovic, N. (2012). Online Map-Matching based on Hidden Markov Model for Real-Time Traffic Sensing Applications. *The 15th International IEEE Conference on Intelligent Transportation Systems*, 117543, 776–781. [http://ieeexplore.ieee.org/xpls/abs\\_all.jsp?arnumber=6338627](http://ieeexplore.ieee.org/xpls/abs_all.jsp?arnumber=6338627)
- Gómez-Sanz, J. J., Fernández, C. R., & Arroyo, J. (2010). Model driven development and simulations with the INGENIAS agent framework. *Simulation Modelling Practice and Theory*, 18(10), 1468–1482. doi:10.1016/j.simpat.2010.05.012
- Gonzales, E., Yang, C., Morgul, F., & Ozbay, K. (2014). *Modeling Taxi Demand with GPS Data from Taxis and Transit*. <http://transweb.sjsu.edu/PDFs/research/1141-modeling-taxi-demand-gps-transit-data.pdf>
- Grau, J. M. S., Moreira-Matias, L., Saadallah, A., Tzenos, P., Aifadopoulou, G., Chaniotakis, E., & Romeu, M. A. E. (2018). Informed versus non-informed taxi drivers: Agent-Based Simulation framework for assessing their performance. *Transportation Research Board 97th Annual Meeting*.
- Grau, J. M. S., & Romeu, M. A. E. (2015). Agent based modelling for simulating taxi services. *Procedia Computer Science*, 52(1), 902–907. doi:10.1016/j.procs.2015.05.162
- Guttman, A. (1984). R-trees: a dynamic index structure for spatial searching. *ACM SIGMOD Record*, 14(2), 47. doi:10.1145/971697.602266
- Hadachi, S. A., & Kibal, J. (n.d.). Implementation of Vector based map- matching algorithm.
- Harvey, A., & Oryshchenko, V. (2012). Kernel density estimation for time series data.



- International Journal of Forecasting*, 28(1), 3–14. doi:10.1016/j.ijforecast.2011.02.016
- HDFS Architecture Guide. (2013). [https://hadoop.apache.org/docs/r1.2.1/hdfs\\_design.html](https://hadoop.apache.org/docs/r1.2.1/hdfs_design.html)
- Helbing, D. (2012). Agent-Based Modeling. In D. Helbing (Ed.), *Social Self-Organization: Agent-Based Simulations and Experiments to Study Emergent Social Behavior* (pp. 25–70). Springer Berlin Heidelberg. doi:10.1007/978-3-642-24004-1\_2
- Horton Data Platform. (n.d.). <https://hortonworks.com/products/data-platforms/hdp/>
- Inman, H. F., & Bradley, E. L. (1989). The Overlapping Coefficient as a Measure of Agreement Between Probability Distributions and Point Estimation of the Overlap of two Normal Densities. *Communications in Statistics - Theory and Methods*, 18(10), 3851–3874. doi:10.1080/03610928908830127
- Intel. (2012). *Intel's 2014 IT Manager Survey on How Organizations Are Using Big data*. Intel IT Center. doi:10.1007/978-3-319-10665-6
- Kanasugi, H., Sekimoto, Y., Kurokawa, M., Watanabe, T., Muramatsu, S., & Shibasaki, R. (2013). Spatiotemporal route estimation consistent with human mobility using cellular network data. *IEEE International Conference on Pervasive Computing and Communications Workshops, PerCom Workshops 2013*, (March), 267–272. doi:10.1109/PerComW.2013.6529493
- Kanungo, T., Mount, D. M., Netanyahu, N. S., Piatko, C. D., Silverman, R., & Wu, A. Y. (2002). An efficient k-means clustering algorithm: analysis and implementation. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 24(7), 881–892. doi:10.1109/TPAMI.2002.1017616
- Katayama, N., & Satoh, S. (1997). The SR-tree: An Index Structure for High-Dimensional Nearest Neighbor Queries. *Proceedings of the 1997 ACM SIGMOD international conference on Management of data - SIGMOD '97*, 369–380. doi:10.1145/253260.253347
- Ke, J., Zheng, H., Yang, H., & Chen, X. (Michael). (2017). Short-term forecasting of passenger demand under on-demand ride services: A spatio-temporal deep learning approach. *Transportation Research Part C: Emerging Technologies*, 85(June), 591–608. doi:10.1016/j.trc.2017.10.016
- Ke, S., Gong, J., Li, S., Zhu, Q., Liu, X., & Zhang, Y. (2014). A hybrid spatio-temporal data indexing method for trajectory databases. *Sensors (Basel, Switzerland)*, 14(7), 12990–13005. doi:10.3390/s140712990
- Kleijnen, J. P. C. (1996). Five-stage Procedure for the Evaluation of Simulation Models Through

- Statistical Techniques. In *Proceedings of the 28th Conference on Winter Simulation* (pp. 248–254). Washington, DC, USA: IEEE Computer Society. doi:10.1145/256562.256616
- Laney, D. (2001). *3D data management: Controlling data volume, velocity and variety*. META Group Research Note (Vol. 949). doi:10.1016/j.infsof.2008.09.005
- Leduc, G. (2008). *Road Traffic Data : Collection Methods and Applications*. EUR Number: Technical Note: JRC 47967 (Vol. JRC 47967). doi:JRC 47967 - 2008
- Li, B., Zhang, D., Sun, L., Chen, C., Li, S., Qi, G., & Yang, Q. (2011). Hunting or waiting? Discovering passenger-finding strategies from a large-scale real-world taxi dataset. *2011 IEEE International Conference on Pervasive Computing and Communications Workshops, PERCOM Workshops 2011*, 63–68. doi:10.1109/PERCOMW.2011.5766967
- Li, Q., Zheng, Y., Xie, X., Chen, Y., Liu, W., & Ma, W. Y. (2008). Mining user similarity based on location history. *Proceedings of the 16th ACM SIGSPATIAL international conference on Advances in geographic information systems*, (c), 34. doi:10.1145/1463434.1463477
- Lindfield, G., & Penny, J. (2017). *An Introduction to Optimization*. *Introduction to Nature-Inspired Optimization*. doi:10.1016/B978-0-12-803636-5.00001-3
- Liu, K., Yamamoto, T., & Morikawa, T. (2006). An Analysis of the Cost Efficiency of Probe Vehicle Data at Different Transmission Frequencies. *International Journal of ITS Research*, 4(1), 21–28.
- Liu, K., Yamamoto, T., & Morikawa, T. (2007). Comparison of time/space polling schemes for a probe vehicle system. *Proceedings of the 14th World Congress on Intelligent Transport Systems*.
- Liu, K., Yamamoto, T., & Morikawa, T. (2008). Study on the cost-effectiveness of a probe vehicle system at lower polling frequencies. *International Journal of ITS Research*, 29–36.
- Lou, Y., Zhang, C., Zheng, Y., Xie, X., Wang, W., & Huang, Y. (2009). Map-matching for low-sampling-rate GPS trajectories. *Proceedings of the 17th ACM SIGSPATIAL International Conference on Advances in Geographic Information Systems - GIS '09*, (c), 352–361. doi:10.1145/1653771.1653820
- Lv, H., Fang, F., Zhao, Y., Liu, Y., & Luo, Z. (2017a). A Performance Evaluation Model for Taxi Cruising Path Recommendation System. In J. Kim, K. Shim, L. Cao, J.-G. Lee, X. Lin, & Y.-S. Moon (Eds.), *Advances in Knowledge Discovery and Data Mining* (Vol. 10235 LNAI, pp. 156–167). Cham: Springer International Publishing. doi:10.1007/978-3-319-57529-2

- Lv, H., Fang, F., Zhao, Y., Liu, Y., & Luo, Z. (2017b). A Performance Evaluation Model for Taxi Cruising Path Recommendation System. In *Advances in knowledge discovery and data mining* (Vol. 10235 LNAI, pp. 156–167). doi:10.1007/978-3-319-57529-2
- Maciejewski, M., Salanova, J. M., Bischoff, J., & Estrada, M. (2016). Large-scale microscopic simulation of taxi services. Berlin and Barcelona case studies. *Journal of Ambient Intelligence and Humanized Computing*, 7(3), 385–393. doi:10.1007/s12652-016-0366-3
- Macqueen, J. (1967). Some methods for classification and analysis of multivariate observations. *Proceedings of the Fifth Berkeley Symposium on Mathematical Statistics and Probability*, 1(233), 281–297. doi:citeulike-article-id:6083430
- MapReduce Tutorial. (2013). [https://hadoop.apache.org/docs/r1.2.1/mapred\\_tutorial.html](https://hadoop.apache.org/docs/r1.2.1/mapred_tutorial.html)
- Mattheis, S., Khaled Al-Zahid, K., Engelmann, B., Hildisch, A., Holder, S., Lazarevych, O., et al. (2014). Putting the car on the map: A scalable map matching system for the Open Source Community. *Lecture Notes in Informatics (LNI), Proceedings - Series of the Gesellschaft für Informatik (GI), P-232*, 2109–2119. <http://www.scopus.com/inward/record.url?eid=2-s2.0-84922567591&partnerID=40&md5=1fa47bf4592182b29baf80e0625c4bf6>
- Miwa, T., Sakai, T., & Morikawa, T. (2004). Route Identification and Travel Time Prediction Using Probe-Car Data. *International Journal of ITS Research*, 2(1).
- Moreira-Matias, L., Gama, J., Ferreira, M., Mendes-Moreira, J., & Damas, L. (2013). On predicting the taxi-passenger demand: A real-time approach. In L. Correia, L. P. Reis, & J. Cascalho (Eds.), *Progress in Artificial Intelligence* (Vol. 8154 LNAI, pp. 54–65). Berlin, Heidelberg: Springer Berlin Heidelberg. doi:10.1007/978-3-642-40669-0\_6
- Moreira-Matias, L., Gama, J., Ferreira, M., Mendes-Moreira, J., & Damas, L. (2016). Time-evolving O-D matrix estimation using high-speed GPS data streams. *Expert Systems with Applications*, 44, 275–288. doi:10.1016/j.eswa.2015.08.048
- Nagashima, Y., Hattori, O., & Kobayashi, M. (n.d.). Improvement of Traffic Signal Control Using Probe Data, 44–47.
- Nam, D., Hyun, K. (Kate), Kim, H., Ahn, K., & Jayakrishnan, R. (2016). Analysis of Grid Cell–Based Taxi Ridership with Large-Scale GPS Data. *Transportation Research Record: Journal of the Transportation Research Board*, 2544(March 2017), 131–140. doi:10.3141/2544-15
- Newson, P., & Krumm, J. (2009). Hidden Markov map matching through noise and sparseness. *Proceedings of the 17th ACM SIGSPATIAL International Conference on Advances in*

- Geographic Information Systems - GIS '09*, 336–343. doi:10.1145/1653771.1653818
- Nutanong, S., Jacox, E. H., & Samet, H. (2011). An incremental Hausdorff distance calculation algorithm. *Proceedings of the VLDB Endowment*, 4(8), 506–517. doi:10.14778/2002974.2002978
- Peungnumesai, A., Witayangkurn, A., Nagai, M., Arai, A., Ranjit, S., & Ghimire, B. R. (2017). Bangkok Taxi Service Behavior Analysis using Taxi Probe Data and Questionnaire Survey. *Proceedings of the 4th Multidisciplinary International Social Networks Conference*, 1–8. doi:10.1145/3092090.3092117
- Pink, O., & Hummel, B. (2008). A statistical approach to map matching using road network geometry, topology and vehicular motion constraints. *IEEE Conference on Intelligent Transportation Systems, Proceedings, ITSC*, 862–867. doi:10.1109/ITSC.2008.4732697
- Ranjit, S., Nagai, M., Witayangkurn, A., & Shibasaki, R. (2017). Sensitivity analysis of map matching techniques of high sampling rate GPS data point of probe taxi on dense open street map road network of Bangkok in a large-scale data computing platform. In *15th International Conference on Computers in Urban Planning and Urban Management*.
- Raychaudhuri, S. (2008). Introduction to monte carlo simulation. *2008 Winter Simulation Conference*, 91–100. doi:10.1109/WSC.2008.4736059
- Raymond, R., Morimura, T., Osogami, T., & Hirose, N. (2012). Map Matching with Hidden Markov Model on Sampled Road Network, (Icpr), 2242–2245. doi:10.0/Linux-x86\_64
- Ren, M., & Karimi, H. A. (2009). A hidden Markov model-based map-matching algorithm for wheelchair navigation. *Journal of Navigation*, 62(3), 383–395. doi:10.1017/S0373463309005347
- Reuillon, R., Schmitt, C., De Aldama, R., & Mouret, J.-B. (2015). A New Method to Evaluate Simulation Models: The Calibration Profile (CP) Algorithm. *Journal of Artificial Societies and Social Simulation*, 18(1), 12. doi:10.18564/jasss.2675
- Rothlauf, F. (2011). Optimization Problems. In *Design of Modern Heuristics* (pp. 7–45). doi:10.1007/978-3-540-72962-4
- Sadahiro, Y., Lay, R., & Kobayashi, T. (2013). Trajectories of Moving Objects on a Network: Detection of Similarities, Visualization of Relations, and Classification of Trajectories. *Transactions in GIS*, 17(1), 18–40. doi:10.1111/j.1467-9671.2012.01330.x
- Salanova, J. M., Romeu, M. E., & Amat, C. (2014). Aggregated Modeling of Urban Taxi Services.

- Procedia - Social and Behavioral Sciences*, 160(Cit), 352–361.  
doi:10.1016/j.sbspro.2014.12.147
- Schroeck, M., Shockley, R., Smart, J., Romero-Morales, D., & Tufano, P. (2012). *Analytics: The real-world use of big data*. IBM Global Business Services Saïd Business School at the University of Oxford.
- Sekimoto, Y., Shibasaki, R., Kanasugi, H., Usui, T., & Shimazaki, Y. (2011). PFlow: Reconstruction of people flow by recycling large-scale fragmentary social survey data. *IEEE Pervasive Computing*, 10(4), 27–35. doi:10.1109/MPRV.2011.43
- Szeto, W. Y., Wong, R. C. P., Wong, S. C., & Yang, H. (2013). A time-dependent logit-based taxi customer-search model. *International Journal of Urban Sciences*, 17(2), 184–198. doi:10.1080/12265934.2013.776292
- Szwed, P., & Pekala, K. (2014). An Incremental Map-Matching Algorithm Based on Hidden Markov Model. *Icaisc*, 8468, 579–590.
- Szymkowiak, A., Larsen, J., & Hansen, L. K. (2001). Hierarchical Clustering for Datamining. *Proceedings of KES2001 Fifth International Conference on KnowledgeBased Intelligent Information Engineering Systems Allied Technologies*, 261–265. <http://eivind.imm.dtu.dk/publications/2001/szymkowiak.kes2001.pdf>
- Tang, J., Jiang, H., Li, Z., Li, M., Liu, F., & Wang, Y. (2016). A Two-Layer Model for Taxi Customer Searching Behaviors Using GPS Trajectory Data. *IEEE Transactions on Intelligent Transportation Systems*, 17(11), 3318–3324. doi:10.1109/TITS.2016.2544140
- Techarattanased, N. (2015). Service Quality and Consumer Behavior on Metered Taxi Services. *International Journal of Economics and Management Engineering*, 9(12), 4107–4111.
- Torrens, P. M. (2010). Agent-based Models and the Spatial Sciences. *Geography Compass*, 4(5), 428–448. doi:10.1111/j.1749-8198.2009.00311.x
- Tu, W., Li, Q., Fang, Z., Shaw, S. lung, Zhou, B., & Chang, X. (2016). Optimizing the locations of electric taxi charging stations: A spatial–temporal demand coverage approach. *Transportation Research Part C: Emerging Technologies*, 65(3688), 172–189. doi:10.1016/j.trc.2015.10.004
- Verbeek, J. J., Vlassis, N., & Kroese, B. (2003). Efficient Greedy Learning of Gaussian Mixture Models. *Neural Computation*, 15(2), 469–485.
- Wang, Y., Zhu, Y., He, Z., Yue, Y., & Li, Q. (2011). Challenges and opportunities in exploiting

- large-scale GPS probe data. *HP Laboratories Technical Report*, (109).  
<http://www.hpl.hp.com/techreports/2011/HPL-2011-109.pdf>
- Weber, M., Liu, L., Jones, K., Covington, M. J., Nachman, L., & Pesti, P. (2010). On map matching of wireless positioning data. *Proceedings of the 18th SIGSPATIAL International Conference on Advances in Geographic Information Systems - GIS '10*, 290. doi:10.1145/1869790.1869832
- Wei, L., Yuhan, X., & Xuerong, X. (2013). Research of indexing techniques for GIS spatial data. In *IEEE Conference Anthology* (pp. 1–4). IEEE. doi:10.1109/ANTHOLOGY.2013.6784989
- White, D. a., & Jain, R. (1996). Similarity indexing with the SS-tree. *Proceedings of the Twelfth International Conference on Data Engineering*, 516–523. doi:10.1109/ICDE.1996.492202
- Witayangkurn, A., Horanont, T., & Shibasaki, R. (2012). Performance comparisons of spatial data processing techniques for a large scale mobile phone dataset. *Proceedings of the 3rd International Conference on Computing for Geospatial Research and Applications - COM.Geo '12*, 1. doi:10.1145/2345316.2345346
- Witayangkurn, A., Horanont, T., & Shibasaki, R. (2013). The Design of Large Scale Data Management for Spatial Analysis on Mobile Phone Dataset. *Asian Journal of Geoinformatics*, 13(3), 17–24.
- Wong, D. W. S., & Huang, Q. (2016). Sensitivity of DBSCAN in identifying activity zones using online footprints. *Proceedings of Spatial Accuracy 2016*, 151–156.
- Wong, K. I., Wong, S. C., Bell, M. G. H., & Yang, H. (2005). Modeling the bilateral micro-searching behavior for urban taxi services using the absorbing Markov chain approach. *Journal of Advanced Transportation*, 39(1), 81–104. doi:10.1002/atr.5670390107
- Wong, R. C. P., Szeto, W. Y., & Wong, S. C. (2014). A cell-based logit-opportunity taxi customer-search model. *Transportation Research Part C: Emerging Technologies*. doi:10.1016/j.trc.2014.08.010
- Wong, R. C. P., Szeto, W. Y., & Wong, S. C. (2015a). Behavior of taxi customers in hailing vacant taxis: a nested logit model for policy analysis. *Journal of Advanced Transportation*, 49(8), 867–883. doi:10.1002/atr.1307
- Wong, R. C. P., Szeto, W. Y., & Wong, S. C. (2015b). A Two-Stage Approach to Modeling Vacant Taxi Movements. *Transportation Research Procedia*, 7(August), 254–275. doi:10.1016/j.trpro.2015.06.014

- Yang, Q., Gao, Z., Kong, X., Rahim, A., Wang, J., & Xia, F. (2016). Taxi operation optimization based on big traffic data. *Proceedings - 2015 IEEE 12th International Conference on Ubiquitous Intelligence and Computing, 2015 IEEE 12th International Conference on Advanced and Trusted Computing, 2015 IEEE 15th International Conference on Scalable Computing and Communications*, 20, 127–134. doi:10.1109/UIC-ATC-ScalCom-CBDCCom-IoP.2015.42
- Yao, H., Wu, F., Ke, J., Tang, X., Jia, Y., Lu, S., et al. (2018). Deep Multi-View Spatial-Temporal Network for Taxi Demand Prediction. *CoRR, abs/1802.0*. <https://arxiv.org/abs/1802.08714v2>
- Yuan, J., Zheng, Y., Zhang, C., Xie, X., & Sun, G. Z. (2010). An Interactive-Voting based Map Matching algorithm. *Proceedings - IEEE International Conference on Mobile Data Management*, 43–52. doi:10.1109/MDM.2010.14
- Yuan, J., Zheng, Y., Zhang, L., Xie, X., & Sun, G. (2011). Where to Find My Next Passenger? *Proceedings of the 13th international conference on Ubiquitous computing (UbiComp'11)*, 109–118.
- Yuan, N. J., Zheng, Y., Zhang, L., & Xie, X. (2013). T-Finder: A Recommender System for Finding Passengers and Vacant Taxis. *IEEE Transactions on Knowledge and Data Engineering*, 25(10), 2390–2403. doi:10.1109/TKDE.2012.153
- Zhang, D., He, T., Lin, S., Munir, S., & Stankovic, J. A. (2016). Taxi-Passenger-Demand Modeling Based on Big Data from a Roving Sensor Network. *IEEE Transactions on Big Data*, 3(3), 1–1. doi:10.1109/TBDDATA.2016.2627224
- Zhang, S., & Wang, Z. (2016). Inferring passenger denial behavior of taxi drivers from large-scale taxi traces. *PLoS ONE*, 11(11), 1–21. doi:10.1371/journal.pone.0165597
- Zhang, Y., & Li, J. (2009). Research and Improvement of Search Engine Based on Lucene. *2009 International Conference on Intelligent Human-Machine Systems and Cybernetics*, 270–273. doi:10.1109/IHMISC.2009.191
- Zhao, Y., Karypis, G., & Fayyad, U. (2005). Hierarchical Clustering Algorithms for Document Datasets. *Data Mining and Knowledge Discovery*, 10(2), 141–168. doi:10.1007/s10618-005-0361-3
- Zheng, Y. U. (2015). Trajectory Data Mining: An Overview. *ACM Transaction on Intelligent Systems and Technology*, 6(3), 1–41. doi:10.1145/2743025
- Zheng, Z., Rasouli, S., & Timmermans, H. (2014). Evaluating the Accuracy of GPS-based Taxi

Trajectory Records. *Procedia Environmental Sciences*, 22, 186–198.  
doi:10.1016/j.proenv.2014.11.019

Zucchini, W. (2003). kernel density estimation. *Applied smoothing techniques Part 1*, (October), 1–20. <http://staff.ustc.edu.cn/~zwp/teach/Math-Stat/kernel.pdf>