Integrate Transportation Planning Models with Machine Learning Algorithms:

A Computational Graph Framework in a Data-Rich Environment

by

Taehooie Kim

A Dissertation Presented in Partial Fulfillment
of the Requirements for the Degree
Doctor of Philosophy

Approved November 2021 by the
Graduate Supervisory Committee:

Ram M. Pendyala, Co-Chair
Xuesong (Simon) Zhou, Co-Chair
Rong Pan
Yingyan Lou

ARIZONA STATE UNIVERSITY

December 2021

ABSTRACT

With the advent of new mobility services and technologies, the complexity of understanding the mobility patterns has been gradually intensified. The availability of large datasets, in conjunction with the transportation revolution, has been increased and incurs high computing costs. These two critical challenges require us to methodologically handle complex transportation problems with numerical performance: fast, high-precision solutions, and reliable structure under different impact factors. That is, it is imperative to introduce a new type of modeling strategy, advancing the conventional transportation planning models.

In order to do this, we leverage the backbone of the underlying algorithm behind machine learning (ML): *computational graph* (CG) and *automatic differentiation* (AD). CG is a directed acyclic graph (DAG) where each vertex represents a mathematical operation, and each edge represents data transfer. AD is an efficient algorithm to analytically compute gradients of necessary functionality. Embedding the two key algorithms into the planning models, specifically parametric-based econometric models and network optimization models, we theoretically and practically develop different types of modeling structures and reformulate mathematical formulations on basis of the graph-oriented representation.

Three closely related analytical and computational frameworks are presented in this dissertation, based on a common modeling methodology of CG abstraction. First, a two-stage interpretable machine learning framework developed by a linear regression model, coupled with a neural network layered by long short-term memory (LSTM) shows the capability of capturing statistical characteristics with enhanced predictability in the context

of day-to-day streaming datasets. Second, AD-based computation in estimating for discrete choice models proves more efficiency of handling complex modeling structure than the standard optimization solver relying on numerical gradients, outperforming the standard methods, Biogeme and Apollo. Lastly, CG allows modelers to take advantage of a special problem structure for the feedback loops, a new class of problem reformulation developed through Lagrangian relaxation (LR), which makes CG based model well suited for reaching a high degree of the integrated demand-supply consistency.

Overall, the deep integration of the practically important planning models with the underlying computationally efficient ML algorithms can enhance behavioral understanding of interactions in real-world urban systems, and the proposed differentiable mathematical structures will enable transportation decision-makers to accurately evaluate different demand-side and supply-side scenarios with a higher degree of convergency and optimality in more complex transportation systems.

DEDICATION

To my mother and father for their love, endless support, and encouragement.

ACKNOWLEDGMENTS

I would first acknowledge to a great number of people who directly and indirectly offered academic advice, professional comments on my research projects, and friendship throughout my doctoral program study at the Arizona State University.

I wish to express gratitude to my co-advisors, Professor Ram M. Pendyala and Professor Xuesong (Simon) Zhou. Without my academic mentors' support, encouragement, and guidance, I would not have been able to reach the completion of this thesis and propose the innovate development for the next generation of transportation modeling.

I would like to further extend my gratitude to Professors Rong Pan and Yinyang Lou, members of my doctoral committee. They provided constructive feedback on my research, extending the perspective of the field of operations research.

Special thanks go to Professor Micheal Blair, who reviewed one of my current research projects carefully and provided truly valuable advice. I would also like to thank Dr. Xin Ye and Dr. Daehyun You, past members of the transportation systems research group at the Arizona State University. They sincerely helped me to learn about the technical requirements for transportation planning projects.

Very special thanks are to my colleagues Dr. Sara Khoeini, Shivam Sharda, Denise Capasso da Silva, Irfan Batur, and Tassio Magassi for their contribution to many research projects and journal papers I have been involved in. I also appreciate Dr. Xin Wu and Jiawei Lu whom I collaboratively worked with to improve programmed structures faster and more effectively.

Finally, I thank my parents, all my friends, and research colleagues who always believe and encourage me in my academic journey, helping me reach out the finish line.

TABLE OF CONTENTS

Page

CHAPTER Page

CHAPTER Page

LIST OF TABLES

LIST OF FIGURES

Figure Page

CHAPTER 1

1. INTRODUCTION

1.1. Motivation

In recent years, the transportation system has been influenced greatly by emerging and disruptive technologies and socio-economic trends. The development of connected and autonomous vehicles (CAVs) and the new mobility services (ride-hailing), provided by transportation network companies (TNCs), have shown significant impacts on individual travel mobility patterns and capacity-limited transportation networks. Uber reported growth in nationwide market share from 9% to 29% whereas existing taxi services observed a decline in their market from 52% to 35% nationwide (Fischer, 2015). More recently, the covid 19 pandemic led to the rapid market growth of app-based food delivery services such as DoorDash or Uber Eats. In response to the influx, transportation agencies have been devoted to the development of modeling frameworks to deeply fathom emerging mobility patterns observed in the ride-hailing market and the app-based digital platforms. Furthermore, with the advent of self-driving vehicles (LeBeau, 2020), travelers now look forward to experiencing/adopting the advanced transportation services, which can potentially promise better transit accessibility and improved driving safety. Such disruption has encouraged scholars to explore the complexity of modeling the upcoming mobility services in built environments and study different transportation planning approaches.

On the one hand, faced with multiple data sources including travel surveys, mobile phone data records, GPS, or sensor data, and the increased bulk of the datasets, the transportation community has actively applied advanced computational methods, machine

learning (ML), to effectively compute the massive data sources and promptly detect unseen trends, helping alleviate traffic issues. Along this line, diverse ML algorithms/methods or deep learning architectures have been constructed to propose real-time management information for large fleets (Safikhani et al., 2017; Lin et al., 2018), find unobserved patterns in data sets (Ashraf et al., 2020), and enhance travel demand models (Wong and Farooq, 2019; Sifringer et al., 2020). The ML-oriented application in the transportation field has demonstrated excellent computational capabilities in enhancing predictive power, analyzing unobserved complex patterns, and dealing with large-scale datasets efficiently, thereby allowing planners and agencies to model patterns of emerging disruptions.

Although the advanced techniques have presented the aforementioned strengths of ML applications and the ability of computational efficiency in the contexts of estimating many model coefficients, as well as the flexibility of constructing complex composite functions, the data-driven methods still need to address challenges in transportation planning: illustration of casual relationships, avoidance of overfitted results, and generation of precise standard error estimates for hypothesis testing. Without meeting the above system-level requirements, it is difficult for researchers and decision makers to deeply understand the human decision-making process (Paredes et al., 2017; Brathwaite and Walker, 2018) and develop collaborative transportation systems. In other words, to ensure the robustness of modeling travelers' behaviors under emerging environments and explore the impact of the disrupted technologies, a mutually beneficial framework of leveraging the advantages of machine learning algorithms and statistically-oriented models is required. In particular, reviewing four pillars of modeling: data, theory, methods, and computing

(Miller, 2020), this research aims to integrate the transportation planning models with machine learning techniques and address the following research questions in this dissertation.

1. How to develop a unified modeling structure to seamlessly integrate traditional statistical inference models and new machine learning techniques?
2. How to adapt automatic differentiation methods (which are building blocks of ML) to greatly enhance the computational efficiency of large-scale econometric model estimation to better uncover complex behavioral parameters and latent variables?
3. How to theoretically define the consistency of transportation demand-side and traffic supply-side models, while providing numerically stable solutions consistent with multi-data sources from real-world systems?

1.2. Objectives

To address the stated challenges and offer general frameworks to integrate machine learning techniques and conventional transportation planning models, the objectives of this dissertation are listed as follows.

1. Formulate and develop a hybrid framework that provides the statistical characteristics with enhanced predictability in the context of day-to-day streaming datasets.
2. Incorporate discrete choice models with machine learning algorithms so as to not only improve computational performance for estimating large parameter sets, but also maintain the interpretability in the conventional models in accordance with the requirement of transportation planning applications.

3. Develop a mathematically rigorous framework capable of integrating travel demand and network models under multiple data sources, thereby ensuring the internal consistency of traveler patterns and traffic flows with hard road capacity constraints.

The first objective is intended to estimate the existing/emerging demand trends and identify the correlation of trends while controlling for other explanatory variables. This conceptual framework will help to resolve a number of challenges: proactively optimizing the usage of large fleets, properly understanding the dynamics of the demand trends, and capturing impactful factors of the patterns. Consequently, this conceptual framework is expected to lead to a systematic strategy for the effective activity demand management (ADM) system.

The second objective aims to establish efficient and extensible computational frameworks for transportation planning models. A flexible modeling framework that can interpret disruptive forces (e.g., automatous vehicles or electrification) will be constructed so that planners can simulate emerging environments. In addition, to promptly update/estimate transportation models involved with high-dimensional survey samples and big data sources, we will propose a new algorithm to formulate analytical models, which can enhance computational performance to process day-to-day streaming datasets in a short time.

The third objective is to introduce an analytical approach to integrate travel demand models and network models. Specifically, to illustrate travel demand dimensions corresponding to physical infrastructure, a new framework for the integrated modeling

structure will be developed. Furthermore, to explore more heterogeneous patterns in demand and supply, we will utilize multiple data sources in the proposed mathematical structure, enhancing accuracy of the model.

The overall objective of this research is to propose three types of approaches to demonstrate the procedure: (1) reconstructing conventional travel demand models, (2) reformulating hierarchical structures of illustrating sequential decision-making, and (3) improving computational efficiency, which enables us to design flexible and extensible frameworks, take advantages of big data sources, and respond to emerging trends in large-scale transportation systems.

1.3. Overview of Proposed Methods

To construct three different methodologies, the backbone of facilitating underlying architectures behind machine learning is incorporated, in terms of a form of *computational graph (CG)*, which is a directed graph capable of defining mathematical formulations using nodes (elementary operations) and edges (directions). It should be noted that automatic differentiation (AD) and backpropagation (BP) which is a special case of reverse mode AD (Kelley et al., 1960; Rall., 1981) are applied to train and optimize graphically-defined structures. With this core algorithmic construct, this research advances transportation planning models by introducing a new set of modeling language, which can be flexible and extendable to interpret disruptive patterns in a statistical manner as well as can handle massive data sources.

Specifically, this paper has focused on three key algorithmic concepts, namely computational graph (CG), backpropagation (BP), and automatic differentiation (AD) to

develop the unified modeling framework, for the innovative combination of conventional transportation models and data-driven architectures. In order to integrate the two different modeling languages, the planning models (econometric models and network optimization problems) are driven by CG. Then, estimating behavioral units and finding optimal traffic flows are solved by the AD algorithm, eventually allowing planners to not only interpret human decision-making process, but also take advantages of machine learning techniques. This new set of modeling tools can further lead to a potential paradigm shift in the field of transportation. For instance, transportation planners can benefit from increased computational efficiency in estimating large-scale models and flexibility in capturing complex interactions in transportation planning.

To validate the advantages of using computational graphs (CGs) in transportation planning applications, three types of CG-based frameworks are introduced: A sequential framework using regression models and deep learning algorithms, econometric models formulated by machine learning algorithms, and integration of travel demand and network models. First, to develop the CG-based stepwise framework, we provide a two-stage interpretable machine learning modeling approach using a linear regression (LR) model and a neural network layered by long short-term memory (LSTM). Through the LR model, we interpret the correlation between explanatory variables and dependent variables. Then, LSTM captures the residuals from the LR model, enhancing predictive performance. To demonstrate the proposed model, New York City (NYC) taxi data is employed where the LR model interprets the correlation of yellow taxi demand and ride-hailing services, and LSTM focuses on capturing hidden patterns from the LR model's residuals. This stepwise

modeling approach will help planners quantify the impact related to new mobility policies while maintaining the robustness of predicting demand.

Second, a computational graph-based framework is applied to the problem of parameter estimation for conventional econometric models and extended versions of choice models. We decompose econometric models with nodes and edges and reformulate them as computational graph-based models to take the full advantages of automatic differentiation (AD). In this paper, we present three different graph-oriented choice models; multinomial logit (MNL), nested logit (NL), and integrated choice and latent variable (ICLV). In order to validate the accuracy of estimated results of the proposed models, we use different datasets including the National Household Travel Survey (NHTS) 2017 dataset, and a synthetic dataset provided by Hess and Palma (2019). Furthermore, the CG-based choice models provide the estimated parameters and associated statistical properties such as standard errors.

Lastly, the integrative approach for capturing complex demand-supply interaction is introduced. This framework can describe human mobility patterns in "modeling" CG nodes and their state transition in CG links, mapping the mobility patterns into physical networks (e.g., mobility service networks) and fully integrate both layers seamlessly. The augmented Lagrangian relaxation (LR) algorithm, burrowed from the field of operations research, is used to solve the consistent representation of the demand-supply modeling structure. Furthermore, instead of finding optimal solutions through the first order methods (e.g., gradient descent) or gradient projection (GP), we implement the BFGS numerical optimizer, quasi-newton method to derive convergence in estimating both demand and

supply models partially through statistical properties such as standard errors. Using an illustrative use case, we combine multiple open-source datasets (e.g., household travel surveys, mobility phone sample data, and GPS) and expect to demonstrate this hierarchical modeling approach can efficiently analyze multi-dimensional (space, time, and states) trajectories and trips, properly forecasting human mobility patterns.

Overall, we establish the innovative modeling method to incorporate machine learning algorithms and the transportation planning models, which can help researchers and decision makers to better quantify the nature of complex transportation systems, specifically human mobility pattern changes and these impact on physical/cyber infrastructures.

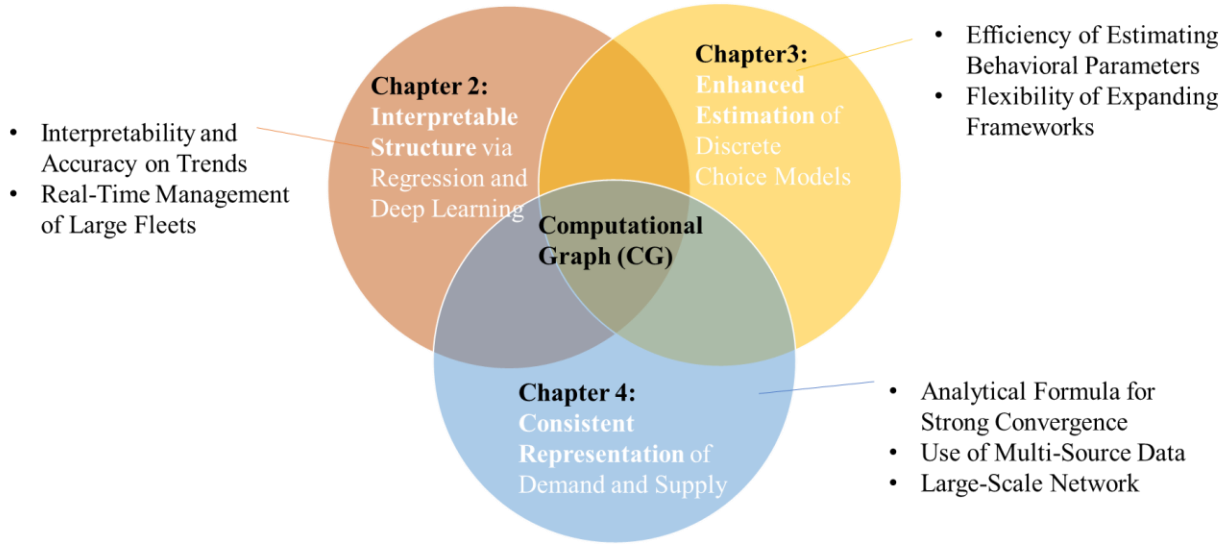


Figure 1.1. Development of Graph-Oriented Transportation Planning-Domain Specific Modeling Language/Platform for Interpretable Learning, Efficient Computation, and Tight Model Consistency

To achieve the proposed goals, this dissertation is organized by five chapters. Chapter 2 develops a stepwise formulation using linear regressions and long-short term

memory (LSTM) which can interpret significant factors affecting existing and emerging mobility demand and predict potential trip demand. Chapter 3 introduces computational graph-based choice models to meet standard requirements associated with statistical analysis and increase computational efficiency in estimating large-scale models. Chapter 4 proposes an analytical mathematical framework to mathematically integrate travel demand and network models and establishes a graphically-oriented architecture to efficiently find optimal traffic flows and traveling costs. Chapter 5 summarizes previous chapters and addresses research contributions and future research plans. Chapter 6 presents prior work cited in this paper. **Figure 1.2.** aims to further illustrate how to extend our proposed methodologies along different research directions. The first axis, *Adaptability & Flexibility*, represents the hybrid modeling structure that can take advantages in the neural network structure and the econometric model to advance transportation planning models. The second axis, *Demand-Supply Coupling*, describes how to effectively design complex modeling structures with a large set of streaming data sources. Lastly, the third axis, *Computing Efficiency*, is advanced computing technologies of GPU and TPU can be applicable, efficiently handing large number of variables and constraints.

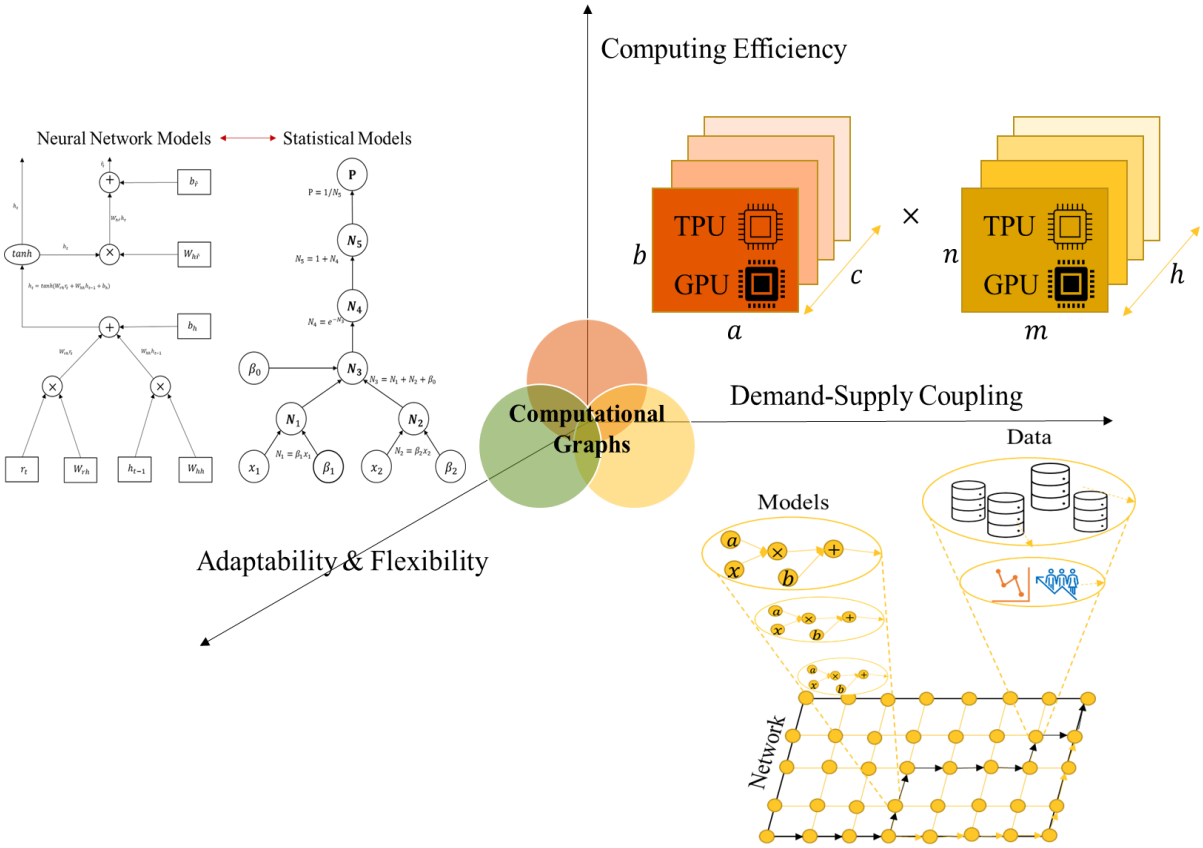


Figure 1.2. Extendibility of Computational Graph-Based Frameworks in Different Transportation Research Directions

CHAPTER 2

2. A STEPWISE INTERPRETABLE MACHINE LEARNING FRAMEWORK BASED ON TIGHTLY COUPLED COMPUTATIONAL GRAPH REPRESENTATIONS

Note: *The substantial part of this chapter is followed by the publication: Kim, T., Sharda, S., Zhou, X. and Pendyala, R.M., 2020. A stepwise interpretable machine learning framework using linear regression (LR) and long short-term memory (LSTM): City-wide demand-side prediction of yellow taxi and for-hire vehicle (FHV) service. Transportation Research Part C: Emerging Technologies, 120, p.102786.*

As app-based ride-hailing services have been widely adopted within existing traditional taxi markets, researchers have been devoted to understand the important factors that influence the demand of the new mobility. Econometric models (EMs) are mainly utilized to interpret the significant factors of the demand, and deep neural networks (DNNs) have been recently used to improve the forecasting performance by capturing complex patterns in the large datasets. However, to mitigate possible (induced) traffic congestion and balance utilization rates for the current taxi drivers, an effective strategy of proactively managing a quota system for both emerging services and regular taxis is still critically needed. This paper aims to systematically design an explainable deep learning model capable of assessing the quota system balancing the demand volumes between two modes. A two-stage interpretable machine learning modeling framework was developed by a linear regression (LR) model, coupled with a neural network layered by long short-term memory (LSTM). The first stage investigates the correlation between the existing taxis and on-demand ride-hailing services while controlling for other explanatory variables. The second

stage fulfills the long short-term memory (LSTM) network structure, capturing the residuals from the first estimation stage in order to enhance the forecasting performance. The proposed stepwise modeling approach (LR-LSTM) forecasts the demand of taxi rides, and it is implemented in the application of pick-up demand prediction using New York City (NYC) taxi data. The experiment result indicates that the integrated model can capture the inter-relationships between existing taxis and ride-hailing services as well as identify the influence of additional factors, namely, the day of the week, weather, and holidays. Overall, this modeling approach can be applied to construct an effective active demand management (ADM) for the short-term period as well as a quota control strategy between on-demand ride-hailing services and traditional taxis.

2.1. Introduction

The emergence of on-demand ride hailing services, such as Uber and Lyft, requires transportation planners to better design comprehensive transportation mobility solutions for informed transportation management and policy making. Uber reported a growth in nationwide market share from 9% to 29% whereas existing taxi services observed a decline in their market from 52% to 35% nationwide (Fischer, 2015). In response to the influx of on-demand services, for instance, New York City Rules announced a protective policy for yellow taxi and green taxi drivers in 2018, and the NYC Department of Transportation (DOT) imposed a congestion pricing, “a new electronic toll in place for drivers entering the busiest stretches of Manhattan” (Mckinley and Hu, 2019). In addition, Sanders and Guse (2019) reported that NYC imposed the new regulations to app-based companies and to protect hardworking drivers in Manhattan: at least 69% of the operating cars in

Manhattan below 96th St. must serve a passenger, and each company needs to set up a fare rate based on the frequency of carrying passengers.

As a result, the exponential growth of this mobility service and its impact on the transportation network lead to intensive research initiatives. In order to interpret the significant factors affecting the ride-hailing demand, diverse statistical modeling approaches were developed and introduced (Lavieri et al., 2017; Lam and Liu, 2018; Gerte et al., 2018). Forecasting approaches were also conducted to optimize the traffic network, dispersing the fleet of the ride-hailing service or yellow taxis within the network (Laptev et al., 2017; Chen et al., 2017; Zhao et al., 2019). Nevertheless, there are still challenges that need further exploration:

(1) Since the numbers of FHV and yellow taxi drivers are closely correlated, if a cap is imposed for a transportation network company (TNC), what is the expected additional demand to be served by, or more precisely, to be shifted to, yellow taxi drivers?

(2) Does the city traffic management agency need to apply a quota regulation during a rainy day or a snowy day?

A concept of the active demand management (ADM) strategy (FHWA, 2012) with respect to on-demand transit has been incorporated to address similar challenges: proactively optimizing the usage of both taxies and on-demand mobility city-wide (e.g., imposing a day-dependent cap for one type of vehicle (Kamga et al., 2015)). In other words, we aim to understand the dynamics of the competing ride hailing services to mitigate the traffic congestion and maintain the balance of the coexisting system for drivers by analyzing the correlation of both emerging and existing mobility services as well as

identifying their own explanatory variables under different situations such as day of the week, holiday, and weather.

The mobility data obtained from NYC is employed as a demonstration use case. A hybrid model developed integrates both multiple linear regression (LR) and long-short term memory (LSTM), a type of recurrent neural network (RNN). The analytical model in the first layer represents available time series data through a number of interpretable parameters. In the second layer LSTM focuses on designing a data-driven approach to account for the hidden pattern from the first layer's residuals. Combining the two-step process, LR-LSTM predicts taxi demand.

The remainder of this paper is organized as follows. Section 2.2 presents the literature review examining prior research that explores statistical models and neural network structures. Section 2.3 describes a real-world data set, namely, New York City (NYC) Taxi Records. In section 2.4, a stepwise framework is proposed to tackle modeling challenges. The application of the integrated (or coupled) model is demonstrated using the real-world dataset, in section 2.5. Lastly, section 2.6 provides conclusion and discusses the estimation results and future work.

2.2. Literature Review

This review section covers two aspects: (1) statistical approaches to understand demand for ride-hailing services and (2) recent modeling means using statistical modeling techniques as well as data-driven approaches. This is followed by a discussion of the motivations behind our proposed integrated approach.

2.2.1. Statistical Models for Demand Analysis of Ride-Hailing Services

Researchers used statistical models to calibrate unknown parameters and further interpret the factors influencing the taxi demand within the service area. In particular, Yang and Gonzales (2014) applied multivariate linear regression to estimate taxi trips by identifying significant explanatory variables, to name a few, population, age, education, income, transit accessibility time, and employment. Safikhani et al. (2017) designed the generalized spatial-temporal autoregressive (STAR) to understand spatial and temporal variations in taxi demand in New York City. Lavieri et al. (2017) utilized a multivariate count approach to study the number of trips generated in a specific zone and observed that different income groups preferred to use the services for different activity purposes. Recently, Lam and Liu (2018) used the discrete choice model to analyze the correlation between dynamic pricing and waiting time in densely populated areas of New York. Gerte et al. (2018) examined the demand for the ride hailing service using a panel based random effects model in order to capture both heteroscedasticity and autocorrelation effects. This study denoted that the highly educated young male group tended to use the ride hailing service frequently.

Overall, the above-mentioned statistical modeling approaches have demonstrated their strength in explaining and predicting the ride-hailing service demand which allows planners to identify significant parameters for informed decision making. Regarding the use of other big data sources, social media data sets have been utilized in a variety of applications to capture individual activity patterns (Gu et al., 2018; Zhang et al., 2017; Hasnat and Hasan, 2018) and detect traffic incidents (Wang et al.,2016; Kuflik et al., 2017). On the other hand, using app-based data (i.e., multi-sourced data with high variances in

both location accuracy and time of travel), He and Shen (2015) and Wang et al. (2019) have proposed conceptual frameworks to estimate the impact of the disruptive mobility services on taxi markets.

Nevertheless, the major challenge of statistical approaches is a lack of predictive accuracy, particularly under a complex data environment with different data sources (Altman et al., 1994; Sarle 1994; Paliwal and Kumar, 2009; Karlaftis and Vlahogianni, 2011). A number of case studies (Kumar et al., 2015; Al-Maqaleh et al., 2016; Golshani et al., 2018; Cui et al., 2018) demonstrated improved prediction accuracy by utilizing the neural network structures compared to statistical models.

2.2.2. Hybrid Approaches Combining Statistical Models with Deep Neural Networks (DNNs)

We first focus on papers in the area of causal inference and prediction, as well as two popular DNN modeling tools using convolutional neural network (CNN) and long short-term memory (LSTM). CNN has demonstrated excellent capabilities in the visual data processing where its structure extracts the image features and classifies the images according to the extracted features (LeCun and Bengio, 1995). On the other hand, LSTM specializes in sequential data processing, and its structure stores significant information and forecasts the sequential data (Hochreiter and Schmidhuber, 1997).

Recently, many studies started to integrate more statistically-oriented modeling features in a deep learning framework. For example, Park et al. (2016) applied Bayesian neural network (BNN) to examine the uncertainty of dependent variables and further interpreted the unknown coefficients in traffic demand prediction models. They used an extracted decision tree approach to provide significant explanatory variables based on a

pedagogical rule extraction algorithm. A study by Ke et al. (2017) applied the random forest framework to select the exogenous variables, ranking these variables' significance. Additionally, they examined the image intensity from map sequences of travel time rates using CNN and LSTM tools. Along this line, Laptev et al. (2017) utilized Bootstrap and Bayesian techniques with LSTM structure to detect abnormality of data and capture irregular patterns.

In the traffic demand estimation area, a recent study by Chen et al. (2017) employed the ensemble learning approach with the RelifF algorithm to identify the important factors. As a result, they were able to explore ride-splitting behavior of on-demand ride services in real-world data sourced from DiDi. More recently, Zhao et al. (2019) examined heterogeneity in mode-switching behavior by proposing a more systematically defined interpretable machine learning approach. To classify the mode-switching behavior, the model-agnostic interpretation tools were used to study the insights on the switching behaviors.

Overall, we observe two emerging research directions in the field of traffic demand modeling. First, researchers have devoted major efforts to develop hybrid modeling frameworks based on statistical and DNN approaches in order to improve the predictive performance. Second, many research teams have focused on developing interpretable models implemented by feature importance and data extraction algorithms to systematically explore the underlying traffic demand behavior and improve the lack of transparency observed in a nested non-linear structure (Samek et al., 2017; Gunning, 2017). Readers interested in the interpretable machine learning techniques can find more details

in prior work (Ribeiro et al., 2016; Lipton, 2018; Molnar, 2020). However, the following challenges still need to be addressed: how to analyze the correlation of both emerging and existing mobility services as well as identify each mode pattern with demand-side factors (i.e., weather, holidays, and day of the week). In order to tackle the challenges, we propose a conceptual interpretable structural decomposition approach as follows:

True demand = Linear explanatory components + Nonlinear pattern + Random fluctuations

To calibrate the different components in this framework, a two-step process is adopted. In the first step the multiple linear regression (LR) captures the primary linear components, and the estimated coefficients offer additional insights on each mobility pattern. The second step utilizes the forecasting capabilities of LSTM network to account for the residuals from the first step.

2.3. Data Description

Multiple data sources are utilized in this study: New York City Taxi & Limousine Commission (NYC-TLC) dataset containing daily mobility demand patterns for the two modes (i.e., For-Hire Vehicles (FHVs) and yellow taxi), weekday/weekend, holidays, and weather information available from National Climate Data Center (NCDC).

2.3.1. New York City (NYC) Taxi Trip Records

The input dataset, NYC-TLC, covers both FHV and yellow taxi trip records collected from 2015 to 2017. In the yellow taxi records the available information includes the pick-up/drop-off events on weekday/weekend, the trip fare, payment type, location ID, and the number of passengers. In the FHV records only the pick-up/drop-off events and location

ID are collected. To set up a fair comparison, we explored the data with an overlapping time period for pick-up/drop-off events. The number of pick-up events is used to investigate the dynamic trip demand with respect to FHV and yellow taxi. The study was done in Murray Hill, Manhattan located in Manhattan below 96th Street, and it is one of the highest taxi pick-up locations: the annual average pick-up demand is approximately 5.7 million, and the total trips made are 17 million over the period of three years.

Figure 2.1 illustrates the volume of pick-ups and trends between yellow taxi and FHV over three years in Murray Hill, Manhattan. The land use of the study area shows a combination of residential and commercial districts. To be specific, Figure 2.1(a) explains the proportion of the annual pick-up volume between the two modes from 2015 to 2017. In addition, Figure 2.1(b) displays the weekly distribution of the pick-up volume between yellow taxi and FHV, where the overall demand gradually increases at the beginning of the week and tapers off as the weekend approaches. To further ensure the length of the data between two modes, we aggregate the event time series data in terms of the daily volume, with the day of the week as a binary indicator.

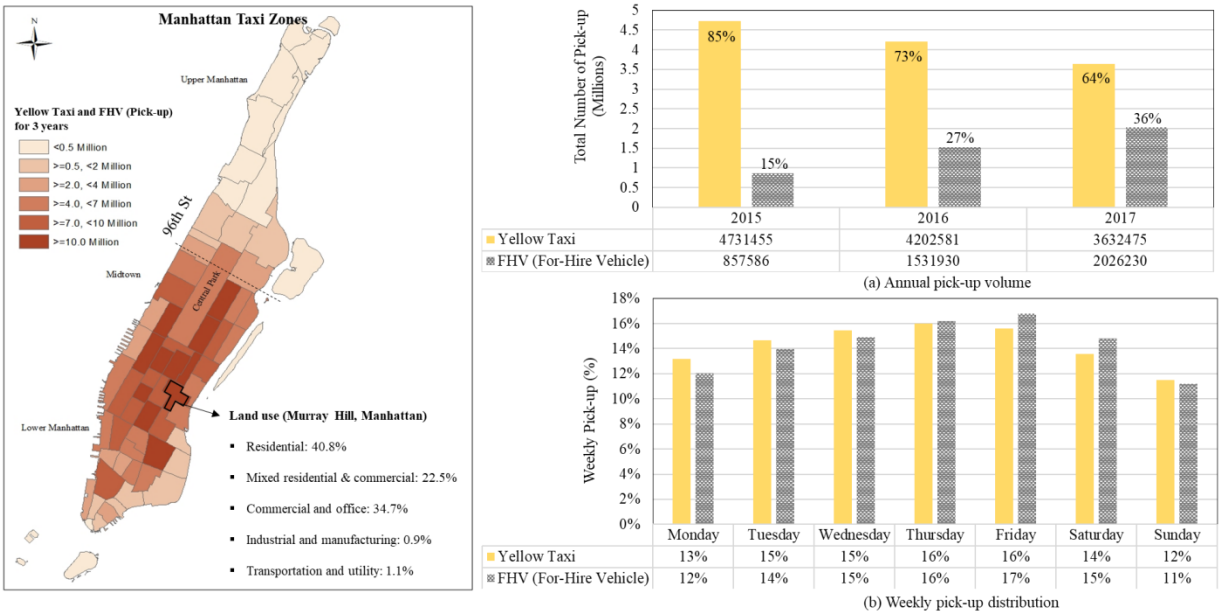


Figure 2.1. Illustrations of Yellow Taxi and FHV Pick-Ups for Murray Hill, Manhattan

Figure 2.2 clearly denotes a mode shifting trend between the yellow taxi and FHV where the transition has stabilized over the years. The percentage of the total trips and the level of usage of yellow taxis have gradually declined as the riders have significantly shifted their demand from yellow taxi to FHV as observed from Figure 2.2(a). The total demand of both modes is in fact, very stable, indicating the ride-hailing service has not further induced new demand for the study area as shown in Figure 2.2(b).

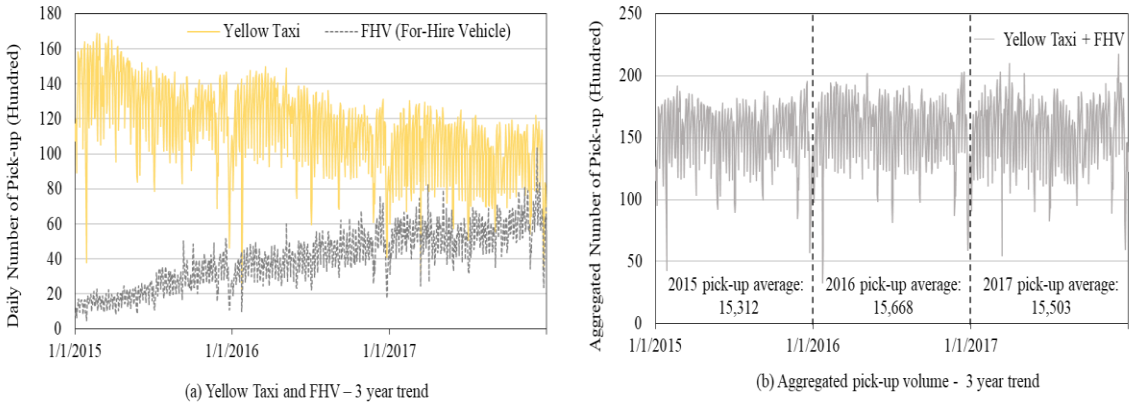


Figure 2.2. Shifts in Pick-Up Demands between Yellow Taxi and FHV over Three Years

2.3.2. Weather and Holidays

The prior study by Schneider (2015) and Guo et al. (2018) reported that the weather condition and holidays affect the usage level of yellow taxi and FHV. Accordingly, information collected by the National Climate Data Center (NCDC) is used in this study, which covers the daily average records of both precipitation (i.e., rain or melted snow) and snow in Central Park, Manhattan. Overall, there are 347 rainy days and 22 snowy days within the investigated years.

In addition, holidays observed by the Federal Holiday calendar are identified in the input data such as New Year's Day, Martin Luther King, Jr. Day, George Washington's Birthday, Memorial Day, Independence Day, Labor Day, Columbus Day, Veterans Day, Thanksgiving Day, and Christmas Day.

2.4. Conceptual Modeling Framework and the Stepwise Procedure

This section describes the system control architecture and the process of the stepwise modeling framework. The architecture is explained in terms of its structural equation formula, and the stepwise process is addressed through multiple linear regression (LR) and Long Short-term Memory (LSTM) neural network.

2.4.1. System Architecture for Predicting Yellow Taxi Demand from FHV Quota to Be Controlled

The problem aims to predict the potential trip demand of yellow taxis during the day, with FHV volume as the exogenous input variable, as a function of the other factors such as weather, holidays, day of the week. Figure 2.3 displays the architecture composed

of the control variables, prediction, and a stepwise approach using two steps (i.e., LR-LSTM). More specifically, we can consider the following use case of the proposed model: the transportation authority plans possible quota for FHV demand, in an effort to create a fair and equitable environment for yellow taxi drivers with reasonable drivers' earning levels.

On day $t-1$, one can use the stepwise model to first estimate yellow taxi demand, briefly denoted as y_{t-1}, using the determined conditions, via LR and then integrate the estimation results with the predicted residuals $\hat{r}_t$ from LSTM, finally forecasting the total pick-up volumes served by yellow taxi, $\hat{y}_t$, at different conditions. On the next day t, we compare the ground truth values with predicted values from our model and then update parameters of LSTM for further use. By setting up a different FHV quota on different days, we hope to meet both system-wide goals of reducing congestion as well as ensuring sufficient taxi utilization rates.

Table 2.1.
Comparison of Characteristics of LR-LSTM with That of Different Time Series Models and Regression

Model	Predictor	Prediction	Quota control	Level of Interpretability	Prediction Accuracy
ARIMA	y_{t-1}	$\hat{y}_t$	No	Medium	Medium to High
LSTM	y_{t-1}	$\hat{y}_t$	No	Low	High
Regression	X_t	$\hat{y}_t$	Yes	High	Low to Middle
LR-LSTM	X_t, r_{t-1}	$\hat{y}_t, \hat{r}_t$	Yes	High, presented in 2.5.1	High, presented in 2.5.2

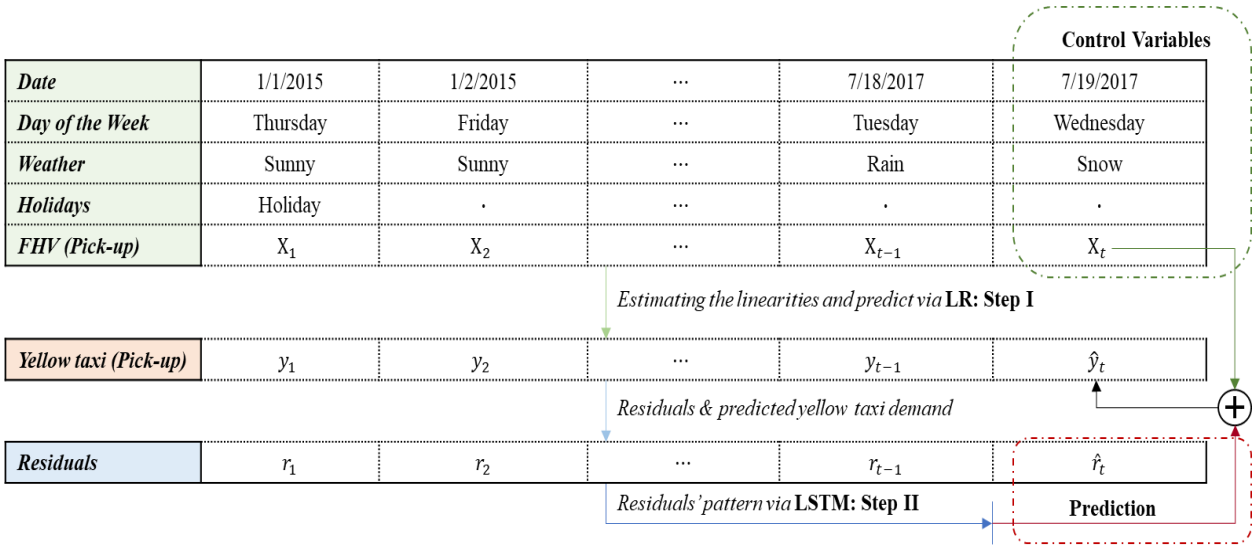


Figure 2.3. Illustration of System Control Architecture to Estimate and Predict Yellow Taxi Pick-Ups

One can further extend this methodology to design personalized incentive schemes in future research, while this study still focuses on controlling the total of FHV or yellow taxi as individual parties. On the other hand, unlike time- series models (e.g., Autoregressive integrated moving average (ARIMA) or long short-term memory (LSTM)) using the historical (correlated) records of yellow taxi as only input data to predict the (unknown) yellow taxi demand, our proposed model seamlessly adds external variables for quota control. Table 2.1 compares the characteristics of the above models. The stepwise framework starting from Step I, the multiple linear regression to Step II, the recurrent neural network (RNN) – LSTM is written as follows:

$$\begin{aligned} &\text{Step I:} && \boldsymbol{y_t} = \boldsymbol{\beta X_t^{\intercal}} \\ &\text{Step II:} && \boldsymbol{\hat{r}_t} = \mathbf{G}(\boldsymbol{\theta}; \boldsymbol{y_{t-1}} - \boldsymbol{\beta X_{t-1}^{\intercal}}) \\ &\text{Prediction:} && \boldsymbol{\hat{y}_t} = \boldsymbol{\beta X_t^{\intercal}} + \boldsymbol{\hat{r}_t} \end{aligned} \tag{2.1}$$

The linearity between trip records of demand y and explanatory coefficients β is captured by $(\beta \mathbf{X}^{\intercal})$, and the residuals r_{t-1}=$(y_{t-1} - \beta \mathbf{X}_{t-1}^{\intercal})$ are fed forward in the residual LSTM function $\mathrm{G}(\cdot)$ to model the nonlinearity effect using the neural network parameters $\boldsymbol{\theta}$. By utilizing both linear and neural network components of β and $\boldsymbol{\theta}$, we can predict the demand of yellow taxi.

Figure 2.4 describes the sequential process of Eq. (2.1) with more details in the following subsection.

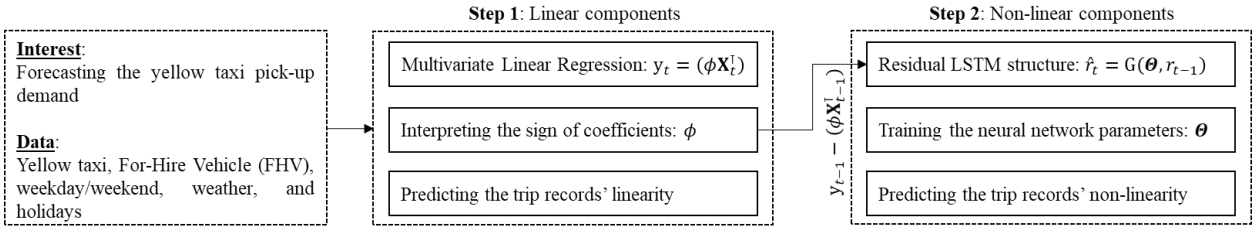


Figure 2.4. Stepwise Calibration Process Using LR-LSTM Modeling

2.4.2. Multiple Linear Regression for Capturing Correlations: LR

To determine the linear correlation between the two modes and the demand-side factors, we now address the pursuit with the regression model, as shown in Eq. (2.2):

$$\hat{y}_{n\times 1} = \mathbf{X}_{n\times p}\hat{\beta}_{p\times 1} = \begin{bmatrix} 1 & \mathrm{X}_{1,1} & \mathrm{X}_{1,2} & \cdots & \mathrm{X}_{1,p-1} \\ 1 & \mathrm{X}_{2,1} & \mathrm{X}_{2,2} & & \mathrm{X}_{2,p-1} \\ & \vdots & & \ddots & \vdots \\ 1 & \mathrm{X}_{n,1} & \mathrm{X}_{n,2} & \cdots & \mathrm{X}_{n,p-1} \end{bmatrix} . \begin{bmatrix} \hat{\beta}_0 \\ \hat{\beta}_1 \\ \vdots \\ \hat{\beta}_p \end{bmatrix} \tag{2.2}$$

$\hat{y}_{n\times 1}$ is the computed trip demand of the yellow taxi and n is the total number of measurements. The matrix $\mathbf{X}_{n\times p}$ indicates the explanatory variables: constant, the actual pick-up volumes of FHV, weekday/weekend, weather, and holidays. p denotes the number of explanatory variables, and $\hat{\beta}_{p\times 1}$ is the calibrated coefficients corresponding to the independent variables.

The ordinary least squares (OLS) method is utilized to calibrate the coefficients: $\hat{\beta}_{p\times 1} = \left(\mathbf{X}_{n\times p}^{\intercal} \cdot \mathbf{X}_{n\times p}\right)^{-1}\mathbf{X}_{n\times p} \cdot y_{n\times 1}$. The residuals $r_{n\times 1}$ between the estimated demand from OLS and the actual demand are calculated as follows, and LSTM extracts possible nonlinear trends of the residuals to improve the predictive accuracy (Goel and Banerjee, 2017).

$$r_{n\times 1} = y_{n\times 1} - \hat{y}_{n\times 1} \tag{2.3}$$

2.4.3. Residual LSTM for Capturing Non-Linear Patterns

This subsection describes LSTM-NN approach (Hochreiter and Schmidhuber, 1997) as an extension version of RNN (Rumelhart et al., 1988), after introducing the principle of RNN and its limitation. Then, the residual LSTM is introduced within a computational graph (CG) framework (Olah, 2015; Baydin et al., 2018; Wu et al., 2018; Sun et al., 2019).

2.4.3.1. Recurrent neural network (RNN)

RNN can be essentially viewed as a non-linear optimization model to minimize the loss function in Eq. (2.4), where the residual term defined in Eq. (2.3) follows time dependent variables.

$$\min L_t = \min \sum_{t=1}^{T} (r_t - \hat{r}_t)^2 \tag{2.4}$$

$$h_t = tanh(W_{rh} r_{t-1} + W_{hh} h_{t-1} + b_h) \tag{2.5}$$

$$\hat{r}_t = W_{h\hat{r}} h_t + b_h \tag{2.6}$$

The defined function $tanh(\cdot)$ in Eq. (2.5) is key to capture non-linear patterns, and Eq. (2.6) can be refined by different structural forms based on specific tasks (e.g., classification or regression). The parameters, W_{rh}, W_{hh}, and $W_{h\hat{r}}$ are applied at every time

step and are shared in the entire structure of RNN. Table 2.2 lists the required components with definitions at time step t. The above optimization model is numerically solved by using the gradient descent algorithm. As discussed in the machine learning community, long data sequences in RNN architecture could lead to the issue of vanishing gradients for the information of updated parameters. That is, if the updated parameters are fractional values, the carried gradients from the long sequence become insignificant. In order to address this issue, the long short-term memory (LSTM) is implemented in this study (Hochreiter and Schmidhuber, 1997).

Table 2.2.
Definitions of Parameters and Variables for RNN Components

RNN at time step *t*	Parameters	Definition
Neural Network (NN) parameters to be estimated	W_{rh}	Weight from input layer to hidden state
	W_{hh}	Weight from previous hidden to current hidden state
	$W_{h\hat{r}}$	Weight from hidden state to output
	b_h	Bias at hidden state
	$b_{\hat{r}}$	Bias at output
Variables	r_{t-1}	Input variable (Residuals from LR)
	h_{t-1}	Hidden variable from the previous time step, *t*-1
	h_t	Hidden variable at the current time step, *t*
	$\hat{r}_t$	Output variable (Prediction)
Composite function	$\hat{r}_t = W_{h\hat{r}}h_t + b_{\hat{r}}$	

2.4.3.2. Long Short-Term Memory (LSTM)

A typical LSTM model consists of four gates (i.e., input, forget, output, and external input gates). The chained structure across the gates transmits not only a hidden state h_t but also previous cell s_t. The shared information of the hidden state and cell state can overcome the vanishing gradient descent effects; the detailed features of LSTM are explained in Olah

(2015) and Goodfellow et al. (2016), and the definition of each parameter used is shown in Table 2.3.

In brief, the first step starts from the forget gate f_t helping to remove the unnecessary information by the sigmoid function $\sigma(\cdot)$, ranging from 0 to 1.

$$f_t = \sigma\left(b_f + W_{rf} r_{t-1} + W_{ff} h_{t-1}\right) \tag{2.7}$$

where b_f is the forget gate bias term. W_{rf} and W_{ff} are weights, h_{t-1} is the previous hidden state, and r_{t-1} is the time series data at current step $t-1$, defined in Eq. (2.3) from the LR model. The second step updates and decides the new information to be stored in the cell state through the input gate i_t and the external input gate g_t.

$$i_t = \sigma(b_i + W_{ri} r_{t-1} + W_{ii} h_{t-1}) \tag{2.8}$$

$$g_t = tanh\left(b_g + W_{rg} r_{t-1} + W_{gg} h_{t-1}\right) \tag{2.9}$$

The input gate i_t determines a gating value ranging from 0 to 1, and a value of 1 means the input information will be fully stored. $tanh(\cdot)$ generates a vector of new candidate values between -1 and 1 guiding the extent of updating the weights in Eq. (2.7) and Eq. (2.8). The multiplication of the input gate i_t and the external gate g_t identifies new significant information, storing it in the cell state, and the third step in Eq. (10) updates the old cell state.

$$s_t = f_t \odot s_{t-1} + i_t \odot g_t \tag{2.10}$$

Both s_t and s_{t-1} denote new and old cell states produced, and $\odot$ is the Hadamard product, the element-wise products of vectors, matrices, or tensors. The final step is proceeded by the output gate o_t.

$$o_t = \sigma(b_o + W_{ro} r_{t-1} + W_{oo} h_{t-1}) \tag{2.11}$$

$$h_t = o_t \odot tanh\,(s_t) \tag{2.12}$$

o_t and h_t are the result of the output gate and a hidden state. Finally, a regression formula defined as the product of the parameter $W_{h\hat{r}}$ and the hidden state h_t with bias term $b_{\hat{r}}$ measures the residual $\hat{r}_t$:

$$\hat{r}_t = W_{h\hat{r}} h_t + b_{\hat{r}} \tag{2.13}$$

In order to compute the approximate residuals, the nonlinear optimization function applied in LSTM is utilized as follows:

$$\min L_t = \sum_{t=1}^{T} \left((W_{h\hat{r}} (o_t \odot tanh(f_t \odot s_{t-1} + i_t \odot g_t)) + b_{\hat{r}}) - r_t \right)^2 \tag{2.14}$$

Eq. (2.14) is the objective function subject to Eqs. (2.7) to (2.13) and LSTM neural network parameters are adjusted to minimize the loss L_t. The adjustment process is proceeded by the *Adam* optimizer, the gradient-based stochastic optimization algorithm proposed by Kingma and Ba (2014) with improved computational efficiency for handling a large data set and parameters. The estimating procedure of the neural network parameters conducts the feedforward and backward propagation process. A computational graph (CG) is shown in Figure 2.5 to illustrate the process in LSTM in which a feedforward step measures the residuals using the defined functions composed of weights and bias, and a backward propagation step minimizes the loss between the computed residuals and the actual residuals by adjusting the parameters.

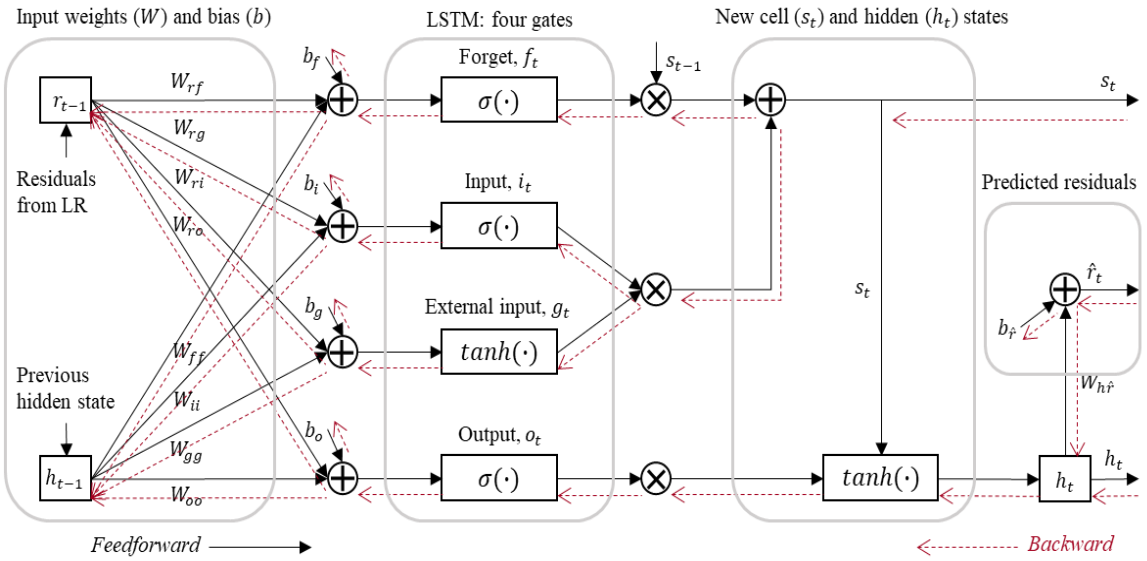


Figure 2.5. Illustration of Inner LSTM Structure: Feedforward and Backward Propagation

Table 2.3.
Definitions of Parameters and Variables for LSTM Components

LSTM at time step t	Terms	Definition
Neural Network (NN) parameters $\boldsymbol{\Theta}$ to be estimated	W_{rf}	Weight from input layer to forget gate
	W_{ff}	Weight from previous forget gate to current forget gate
	W_{ri}	Weight from input to input layer gate
	W_{ii}	Weight from previous input gate to current input gate
	W_{rg}	Weight from input to external input gate
	W_{gg}	Weight from previous external gate to current external gate
	W_{ro}	Weight from input to output gate
	W_{oo}	Weight from previous output gate to current output gate
	$W_{h\hat{r}}$	Weight from hidden state to output
	b_f	Bias at forget gate
	b_i	Bias at input gate
	b_g	Bias at external input gate
	b_o	Bias at output gate
	$b_{\hat{r}}$	Bias at output
Variables	r_{t-1}	Input variable (Residuals from LR)
	h_{t-1}	Hidden variable from the previous time step, t-1

	h_t	Hidden variable at the current time step, *t*
	s_{t-1}	Cell variable from the previous time step, *t*-1
	s_t	Cell variable at the current time step, *t*
	$\hat{r}_t$	Output variable (Prediction)
Composite functions	$\hat{r}_t = W_{h\hat{r}}(o_t \odot tanh(f_t \odot s_{t-1} + i_t \odot g_t)) + b_{\hat{r}}$	

2.4.4. Integrated *M*odel on *T*rip *D*emand *F*orecasting: LR-LSTM

Now we start examining the proposed integrated model LR-LSTM. The linear regression results explain the linearity pattern from variables (i.e., day of the week, holiday, weather, and FHV) so that the pattern learned from the model can help capture the proportional trend on two modes. The residual LSTM generates the unparameterized function to handle the left-over residuals. The mathematical form of the coupled architecture of forecasting the demand is written as:

$$\hat{y}_t = \hat{\beta}\mathbf{X}_t^{\mathrm{T}} + W_{h\hat{r}}(o_t \odot tanh(f_t \odot s_{t-1} + i_t \odot g_t)) + b_{\hat{r}} \tag{2.15}$$

That is, the yellow taxi demand is now estimated by FHV, and the other explanatory variables, together with the predicted residuals. This integrated model can use the regression model to control the pick-up volume of FHV to balance the demand for yellow taxis. Also, the low prediction results caused by traditional extrapolation method can be improved by the residual-oriented model.

Please note that, the input data used for estimating the trip demand is composed of binary variables such as weekday, weekend, and holiday, and the rest of the variables (i.e., FHV, yellow taxi, precipitation, and snow) are continuous, with FHV and yellow taxi values being normalized. In addition, to test the integrated model, the entire dataset is split into the training data, the validation data, and test data as shown in Figure 2.6. Using 800 days (85%) out of 931 days, the hyperparameters used in LSTM are tuned. Then, the

underlying hyperparameters are determined based on the validation experiments shown in Eq. (2.14); the number of hidden units used was 10 to 12, the selected batch size was 10, the number of iteration times, training epoch, was up to 500, the learning rate of the *Adam* optimization was 0.001, and the input dimension was 8 (i.e., weekday/weekend and the prior pickup volume residual). It should be noted that if another set of explainable data is available for the model every day, then we re-execute the tuning process and updating the hyperparameters with the corresponding data-receiving frequency.

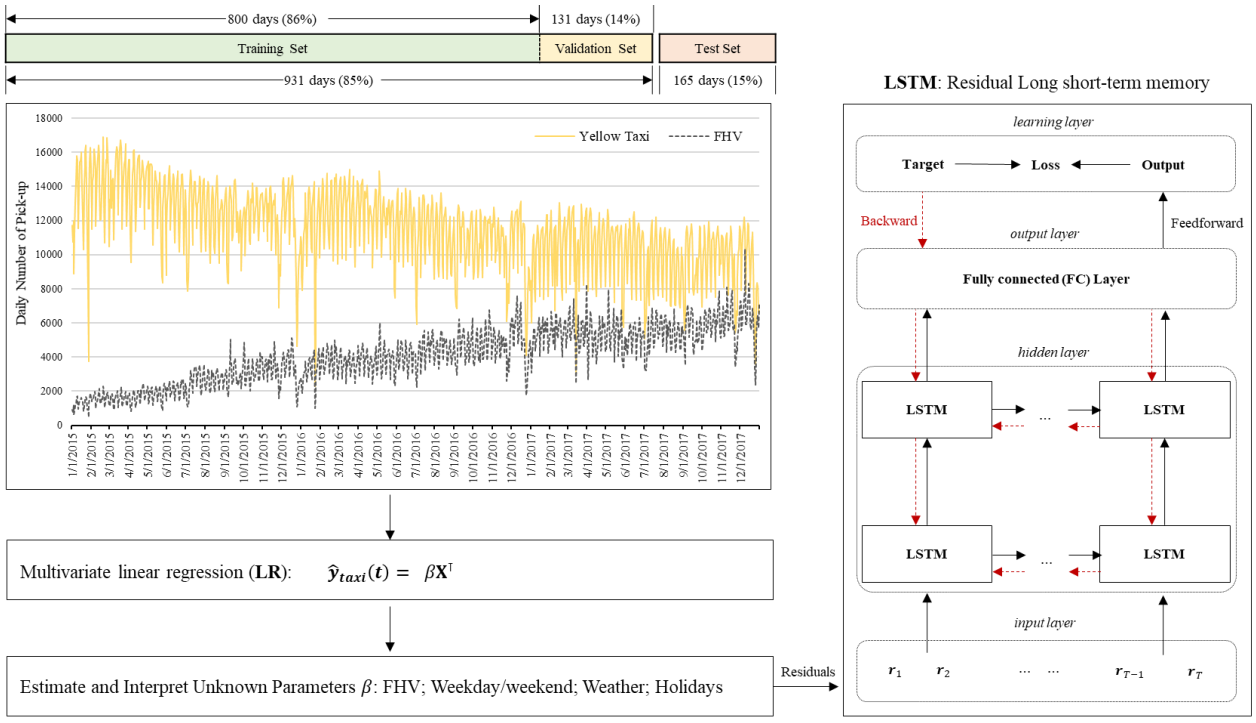


Figure 2.6. Hybrid Model Architecture: Two-Step System Using NYC Taxi Records

The right end side of Figure 2.6 visualizes the two-step process of the coupled LR and LSTM. The *input layer* takes the residual data and transmits the data into the *hidden layer* at the feedforward step denoted as the straight-line arrows in LSTM. Then, a two-layered LSTM expressed in the hidden layer combines the data with the weighted

parameters and bias values based on a basic arithmetic operation (e.g., multiplication and addition) and sends the driven values to the *output layer*. In this layer, the fully connected (FC) layer is added to arrange the output results as a one-dimensional structure, extracting the last output value. Lastly, using the mean squared error (MSE) function between the output and target data given by the *learning layer*, LSTM is trained and updated iteratively through backpropagation expressed as the dotted line arrows. The proposed structure is coded by TensorFlow developed by Abadi et al. (2016). To compare the performance to different time-series models, the results of RMSE (Root Mean Squared Error) and MAPE (Mean Absolute Percentage Error) are detailed in the following section.

$$\begin{aligned} \text{RMSE} &= \sqrt{\frac{1}{T}\sum_{t=1}^{T}(r_t - \hat{r}_t)^2} \\ \text{MAPE} &= \frac{1}{T}\sum_{t=1}^{T}\frac{|r_t - \hat{r}_t|}{r_t} \times 100 \end{aligned} \tag{2.16}$$

where r_t is the residuals at time t sent from LR model, and $\hat{r}_t$ is the predicted value by LSTM. T is the total length of the test dataset (i.e., 165 days for the test set).

2.5. Model Estimation Results

This section examines the estimation and forecasting results for the pick-up demand. Specifically, the first section discusses individual and collective effects of different factors through the parameters estimated by LR. The second section then focuses on LSTM model validation, a sensitivity analysis of LR-LSTM, and the prediction results conducted by the coupled model. Lastly, we examine the forecasting capability of our proposed integrated model, compared to standard alone models such as regression, ARIMA, and LSTM.

2.5.1. Trip demand estimation: LR as Local Interpretable Model

The LR model estimates the regular patterns of the yellow taxi and FHV demand, and the results of LR are presented in Table 2.4. When estimating the yellow taxi demand, the parameter estimates, associated with calendar week, snowy day, holidays, and FHV trips are statistically significant. On the other hand, the estimated FHV demand shows the statistical significance on the aforementioned coefficients as well as rainy day. In addition, the tested variance inflation factor (VIF), a way of measuring multicollinearity, shows a low correlation between independent variables. As shown in Sheather (2009), the generally acceptable range of VIF is less than 5.

According to Table 2.4, the coefficients corresponding to weekday display the positive sign, indicating that customers are more likely to ride yellow taxis or FHV during the day. On the other hand, as the weekend approaches, the pick-up demand volume of two modes decreases. The estimated day-specific coefficients with respect to Tuesday and Saturday are eliminated due to the statical insignificance and the multicollinearity effect. The model-based calibration findings reflect the expected characteristics of the current data set in Figure 2.1(b). The weather conditions affect the level of the taxi and ride-hailing service usage patterns, particularly under a snow event. The negative sign associated with this parameter indicates that the utilization rate of the rides is lower than the regular day. Under the rainy condition, this explanatory factor does not influence the pick-up volume of the taxi statistically. On the one hand, this factor increases the pick-up volume of the ride-hailing services. This noticeable pattern illustrates that travellers prefer to choose the convenient mobility (i.e., app-based ride services) and wait for the service in buildings without being drenched in rain. The coefficients of FHV and yellow taxi can be determined

by switching dependent variables in LR. Both coefficients have the expected negative sign, indicating the inverse relationship with the dependent variables used. This interesting finding explains the gradual decline of the yellow taxi demand, whereas FHV demand follows the opposite pattern, and the numerical difference of the coefficients indicates how much the demand of two modes would be changed when increasing the value of either FHV or yellow taxis.

In Table 2.4, the number of observations is 931 days ranging from 1/1/2015 to 7/21/2017. The goodness of fit measure R^2 denotes that the regression model can explain the variation for the response variable, in terms of an approximate rate at about 57% and 37%. In order to validate if the added predictors truly enhance the interpretability of the model, the adjusted R^2 is measured at 0.565 and 0.366. As the LR model can explain the yellow taxi demand better, we employ the estimated coefficients with respect to the taxi, forecasting the linear pattern. Overall, LR derives the extrapolation results of the yellow taxi volume and then obtains the residuals between the observed data and the estimated results. The statistics and the shape of the residuals derived are plotted in Figure 2.7 with the distributed residuals quantified by μ_Y as the mean of residuals and the variance as σ_Y. As noticed in the figure, the distribution of the residuals generally follows a bell-shaped curve similar to the normal distribution.

Table 2.4.
Multiple Linear Regression (LR) Estimation Results for **Yellow Taxi** and **FHV** Demand

Dependent Variable: **Yellow Taxi**			*Dependent Variable:* **FHV**		
Predictors	*Coefficients (t-statistics)*	VIF	*Predictors*	*Coefficients (t-statistics)*	VIF
Constant	0.7640 (64.337)	-	Constant	0.9094 (30.62)	-
Mon	-0.0362 (-2.89)	1.786	Mon	-0.0949 (-5.7)	1.343
Tue	0.0518 (4.20)	1.720	Tue	-	-

Wed	0.0951 (7.71)	1.725	Wed	0.0663 (4.011)	1.331
Thu	0.1515 (7.71)	1.729	Thu	0.1415 (8.375)	1.392
Fri	0.1445 (11.64)	1.744	Fri	0.1535 (9.169)	1.367
Sat	-	-	Sat	-	-
Sun	-0.1754 (-13.9)	1.783	Sun	-0.2367 (-13.)	1.534
Snow	-0.8224 (-10.1)	1.021	Snow	-1.2091 (-9.1)	1.161
Rainy day	-	-	Rainy day	0.2334 (-10.1)	1.158
Holiday	-0.3401 (-14.2)	1.058	Holiday	-0.3857 (4.8)	1.119
FHV	-0.3289 (-17.9)	1.155	Taxi	-0.7628 (-18.)	1.713
Goodness of fit (number of observations: 931 days)			*Goodness of fit (number of observations: 931 days)*		
R^2(Adj.R^2)	0.573 (0.57)		R^2 (Adj.R^2)	0.372 (0.37)	

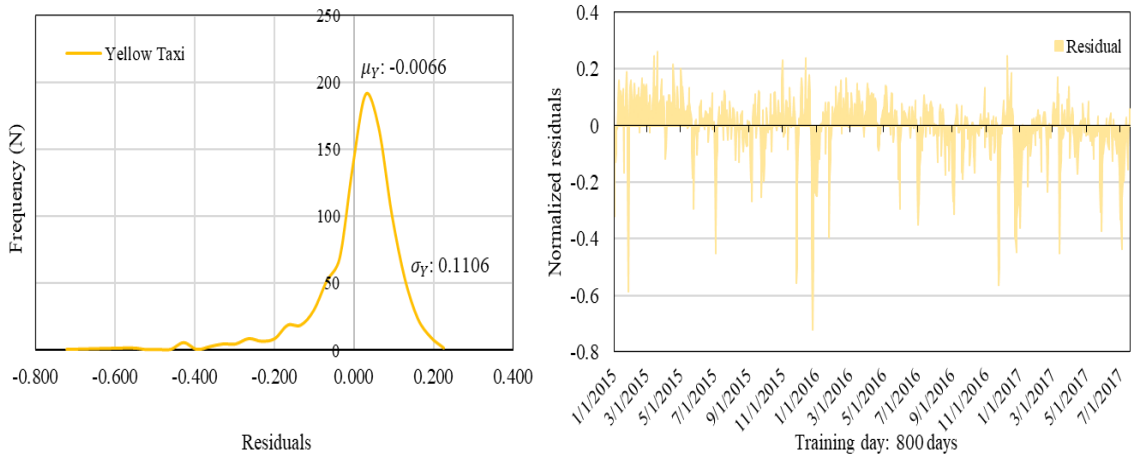


Figure 2.7. Yellow Taxi Residual Distribution and Measured Residuals at Training days

2.5.2. Validation and Prediction: LR-LSTM

Figure. 2.8(a) displays the learning process of the residual patterns and its evolution of the loss function employing the validation dataset with different prediction time steps (e.g., "step 1" indicates the one-step ahead prediction that forecasts the next one day). As noticed, the lowest error, RMSE, appears in the one-step short-term prediction ("step 1"). In addition, the evolutionary error falls rapidly after 10 training epochs, showing a high

convergence speed of the model. Figure 2.8(b) further performs the sensitivity analysis for the proposed LSTM model. The test set error measured by RMSE gradually increases when the model predicts longer than the one-step forecasting period. The baseline of RMSE observed in "step1" is 0.1129, and the seven-step "medium term" prediction denotes that the error increases approximately 48% from the baseline.

Using the multiple linear regression estimators and the residuals measured by LSTM, the number of trips served by yellow taxis at Murray Hill in Manhattan is forecasted in Figure 2.8(c). The test dataset covers the time period from 7/22/2017 to 12/31/2017 (i.e., 165 days), predicting the following day. The predicted demand is plotted along the y-axis as the normalized values with the look-ahead window of prediction on the x-axis. Specifically, Figure 2.8(c) describes the forecasts of the yellow taxi's daily demand that follow the test pattern (i.e., true records) properly across the entire time horizon. This proposed process can be applicable in predicting the demand of the FHV volume by switching the identification of independent and dependent variables.

2.5.3. Model Performance Comparison

In this section we provide a comparison of the proposed model with the different time-series modeling approaches under three different experiments predicting morning peak-hour volumes collected from 6AM to 9AM, evening peak-hour volumes recorded from 4PM to 8PM, and daily volumes of yellow taxis. Evaluating the performance errors (i.e., RMSE and MAPE), we examine the accuracy of LR-LSTM. Autoregressive integrated moving average (ARIMA) and LSTM models are representative of the performance benchmark. The measured RMSE and MAPE presented in Table 2.6 demonstrates the

prediction capability of LR-LSTM. This experiment is prepared by the hold-out strategy (Refaeilzadeh et al., 2009). Using the training and validation dataset, hyperparameters of each model were tuned in decreasing the error between the observed data and the estimated results.

We examine the performance of forecasting the daily volume of yellow taxis at first. According to the validation result (i.e., 931 days), LR-LSTM displays the lowest RMSE and MAPE values compared to other candidates such that the proposed model shows a better performance on the test dataset (i.e., 165 days); an improvement of 34.2% w.r.t MLR, 19.5% w.r.t ARIMA, and 27.4% w.r.t univariate LSTM. On the other hand, the prediction results of multivariate LSTM are built on the basis of not only the historical yellow taxi data but also the explanatory variables such as weekday/weekend, holidays, and weather. It should be remarked that, although multivariate LSTM (M-LSTM) has an RMSE 4% lower than the LR-LSTM result, M-LSTM does not carry a clear interpretable model structure which prevents it from being used effectively in the active demand management application. In addition, the prediction error of the proposed LR-LSTM model is 15.04% in terms of MAPE, which is consistently lower than the the other models including ARIMA, univariate LSTM and within a similar range as multivariate LSTM.

In order to test whether the developed model is able to forecast well under possible oversaturated conditions (that is passenger demand larger than supply), we also examine the predicted results of the morning and evening peak-hour volumes in Table 6. For the morning peak-hour prediction, LR-LSTM presents the second-best performance on the validation dataset (931 days) as well as on the test dataset (165 days). Interestingly, the

multivariate LSTM structure shows the best performance for the validation dataset, but the univariate LSTM model derives the lowest RMSE in the test dataset. In other words, LR-LSTM can maintain the robustness in forecasting validation and test datasets of the morning peak-hour volumes. However, due to the fact that the datasets still include a high variance and other unobserved factors, LR-LSTM and benchmark models display high MAPE errors. For instance, multiple linear regression (MLR) indicates MAPE of 50%. For the evening peak-hour prediction, LR-LSTM on the other hand demonstrates the robustness of predicting the evening peak-hour volumes and the measured RMSE and MAPE present the second-best result on both datasets. The parameter configuration of studied models is listed as follows:

(a) MLR (multiple linear regression): The measured performance is only based on the extrapolation. The prediction on the yellow taxi trips is implemented using the defined coefficients in Table 2.4 and datasets consisting of the FHV trips, day of the week, weather, and holidays. Similarly, using coefficients estimated by the datasets of the morning and evening peak-hour volumes of the medallion taxi, MLR forecasts the taxis demand and proposes RMSE and MAPE shown in Table 2.6.

(b) ARIMA: To determine ARIMA model parameters for the morning and evening peak-hour, and the daily demand, the autocorrelation function (ACF) and the partial autocorrelation function (PACF) proposed by Box et al. (2015) are used to construct our ARIMA (2, 1, 1) model for the prediction of the daily volume. That is, two number of lag observations (AR), one number of times called the degree of difference (I), and the one size of moving average window (MA) are used in our experiment. The

constructed ARIMA (2, 1, 1) model is represented as follows:

$$\hat{y}_t = y_{t-1} + \alpha_1(y_{t-1} - y_{t-2}) + \alpha_2(y_{t-2} - y_{t-3}) + \beta_1(\varepsilon_{t-1}) + constant \tag{2.17}$$

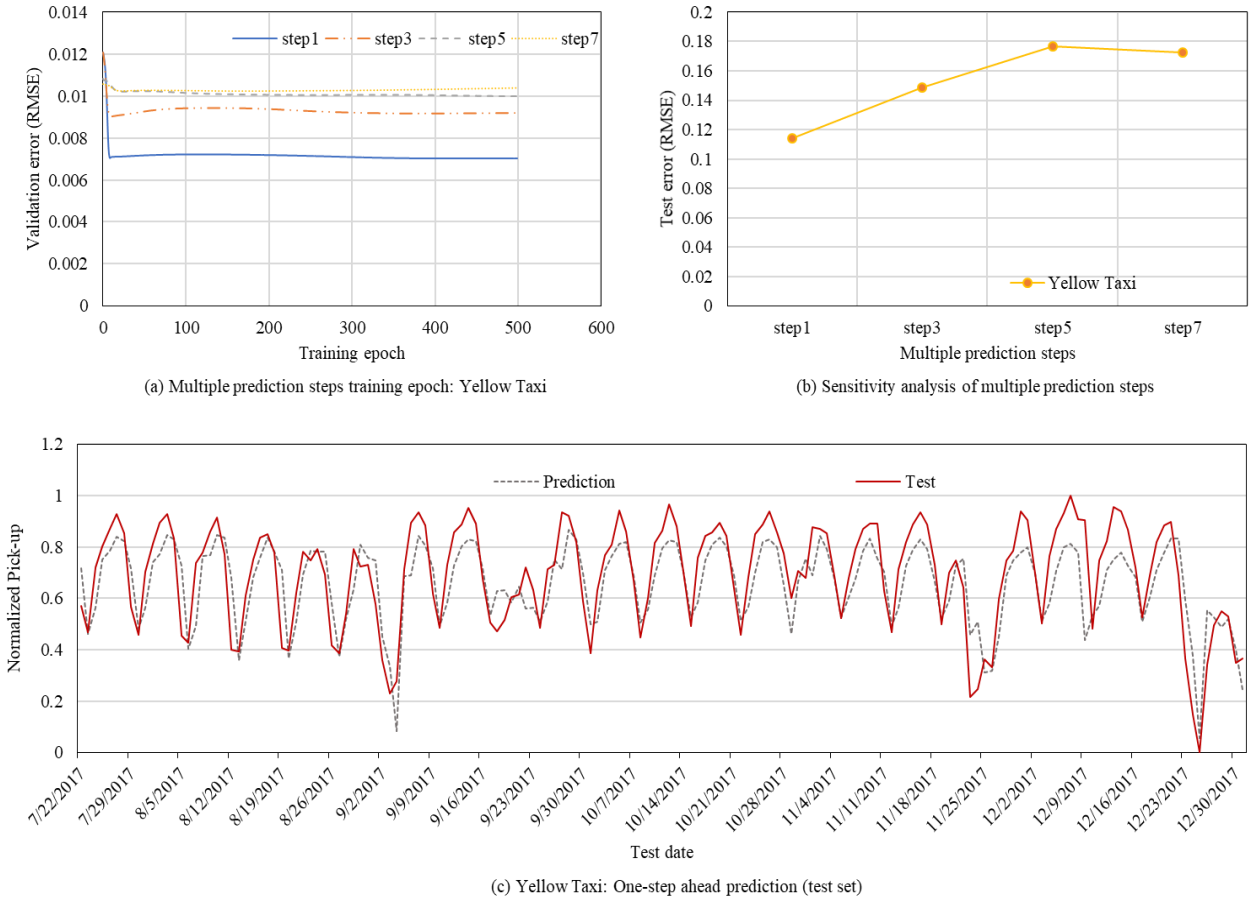
where the predicted yellow taxi demand is calculated by three observed data, namely, y_{t-1}, y_{t-2}, and y_{t-3} with the estimated parameters and constant; both α_1 and α_2 are the AR coefficients, and β_1 is the MA coefficient. Also, the term, ε_{t-1}, indicates the residuals between the observed data and the predicted result, $y_{t-1} - \hat{y}_{t-1}$. Also, in the same manner, the order of the ARIMA model is defined by the proposed method such that the order of ARIMA to forecast the morning peak-hour demand follows ARIMA (2, 1, 3), and the defined configuration of ARIMA for predicting the evening peak-hour is ARIMA (3, 1, 4). The detailed description of calibrated coefficients and the goodness of fit are shown in Table 2.5. The model information-related criteria, AIC and BIC, informs that the lowest values of AIC/BIC are found in the daily volume prediction of yellow taxis, showing the lowest values of RMSE and MAPE in Table 2.6.

(c) LSTM: Two different input features are considered for LSTM. Univariate LSTM is trained by the previous time steps of the yellow taxi only. For instance, the taxi demand for 7/20/2017 was only predicted by 7/19/2017 data. Multivariate LSTM employs weekday/weekend, holiday, weather, and the historical time steps of the taxi to train the model. Both LSTM model structures for predicting the daily taxi demand are configured as: 10-12 hidden units are used, the learning rate is set to 0.001, the training epoch covers 500 steps, and the network structure is one input layer, two hidden layers (LSTM), fully connected layer, and output layer. On the one hand, as the time series

data (i.e., morning and evening peak-hour volumes of yellow taxis) has the high variance, input data structures of training univariate and multivariate LSTM are adjusted. In other words, instead of predicting the demand for 7/20/2017 by the previous day data, it is predicted by the seven days of the yellow taxi demand (7/13/2017 to 7/19/2017). With the adjusted input data structure, we determine different configurations for both LSTM models: 15-17 hidden units are employed, the training epoch is between 500 to 1000 steps, and the network structure is similar to the previous network structure except for the number of stacked LSTM layers; 3-4 stacked layers are examined.

Table 2.5.
ARIMA Estimation Results for Yellow Taxi Demand

	ARIMA (2, 1, 1)	**ARIMA (2, 1, 3)**	**ARIMA (3, 1, 4)**
Predictors	*Estimated coefficients (t-statistics)*		
Constant	-0.0003 (-1.841)	-0.0002 (-9.416)	-0.0001 (-2.222)
α_1 (Auto Regressive)	0.8095 (26.452)	1.2386 (174.764)	0.8053 (5.058)
α_2 (Auto Regressive)	-0.4119 (-13.521)	-0.9895 (-108.529)	-0.4519 (-2.274)
α_3 (Auto Regressive)	-	-	-0.4358 (-2.713)
β_1 (Moving Average)	-0.9706 (-87.491)	-2.2235 (-142.36)	-1.9283 (-11.299)
β_2 (Moving Average)	-	2.1683 (54.43)	1.5394 (4.037)
β_3 (Moving Average)	-	-0.9448 (-38.402)	-0.3166 (-0.823)
β_4 (Moving Average)	-	-	-0.2832 (-1.656)
	Goodness of fit (number of observations: 931 days)		
Fit Criteria	**Daily**	**Morning Peak-hour**	**Evening Peak-hour**
AIC	-1724.612	-869.926	-1189.187
BIC	-1700.436	-836.079	-1145.670



(a) Multiple prediction steps training epoch: Yellow Taxi

(b) Sensitivity analysis of multiple prediction steps

(c) Yellow Taxi: One-step ahead prediction (test set)

Figure 2.8. Training Validation, Sensitivity Analysis, and Estimated Prediction of Yellow Taxi Pick-Up Demand

Table 2.6.
Comparisons of One-Step ahead Validation and Prediction Performance w.r.t Yellow Taxi Demand

Daily	**Validation Dataset (931 days)**		**Test Dataset (165 days)**	
	RMSE	**MAPE (%)**	**RMSE**	**MAPE (%)**
MLR	0.0998	14.54	0.1715	24.03
ARIMA	0.1114	15.47	0.1404	19.52
Univariate LSTM	0.1110	17.52	0.1556	22.85

Multivariate LSTM	0.0738	9.77	0.1081	14.26
LR-LSTM	0.0620	8.69	0.1129	15.04
Morning (6-9AM)	**Validation Dataset (931 days)**		**Test Dataset (165 days)**	
	RMSE	**MAPE (%)**	**RMSE**	**MAPE (%)**
MLR	0.1886	37.58	0.2058	50.63
ARIMA	0.1608	32.74	0.1791	46.67
Univariate LSTM	0.1480	29.25	0.1499	37.42
Multivariate LSTM	0.1304	22.82	0.1766	41.97
LR-LSTM	0.1478	29.57	0.1655	39.98
Evening (4-8PM)	**Validation Dataset (931 days)**		**Test Dataset (165 days)**	
	RMSE	**MAPE (%)**	**RMSE**	**MAPE (%)**
MLR	0.1240	22.21	0.1237	27.25
ARIMA	0.1272	22.73	0.1714	38.83
Univariate LSTM	0.1300	23.84	0.1233	27.14
Multivariate LSTM	0.1135	20.78	0.1217	27.10
LR-LSTM	0.1189	20.99	0.1228	27.31

2.6. Summary and Conclusions

There are many studies using machine learning models for estimating ridership trends. However, the lack of ability, commonly observed in those data-fitting oriented models, of sensing and understanding the significantly influential factors for transportation service demand could cause difficulties in constructing proper ADM policies and decision tools. This paper proposes a hybrid modeling framework, LR-LSTM, to facilitate the planning effort for balancing the utilization rates of the emerging FHV service and regular taxis. Our developed model formulation integrates the multiple linear regression (LR) and LSTM to

forecast the daily and peak-hour taxi demand. With different data sources being used as explanatory variables such as FHV, weekday/weekend, snow/rain, weather conditions, and holidays, LR is employed to select the statistically important variables and interpret the correlation between the variables. LSTM helps improve the prediction accuracy by capturing the variance that the extrapolation model cannot measure.

Essentially, the proposed methodology shows the capability of maintaining the interpretability as well as predictability, indicating the positive potential of achieving the effective active demand management (ADM) system within a city-wide network. To our limited knowledge, this study represents the first integrated modeling effort of the quota control applications between on-demand ride-hailing services and regular taxis, through a combination of the econometric formulation with the deep learning structure. This methodology is validated based on the real-world experiment obtained from New York City Taxi & Limousine Commission (NYC-TLC), particularly forecasting the pick-up demand of the yellow taxi in Murray Hill, Manhattan. The results of RMSE and MAPE outperform the multiple linear regression, ARIMA, and univariate LSTM fed exclusively with historical records. With the well calibrated LR-LSTM tool, we can examine different scenarios for policy analysis. For instance, if a policy starts regulating the FHV pick-up demand in New York City, this approach can estimate the impact of the regulation within the network. Overall, by setting up different FHV quota on different days, we can accomplish both system-wide goals of reducing congestion as well as ensuring sufficient

taxi utilization rates. More detailed future research along this study can be built on the following premises in the current paper.

(1) A stepwise modeling framework could further refine coupled modeling systems to balance the usage level of the yellow taxi demand and for-hire vehicle (FHV) in the coexisting environment with public transportation systems, so that a proper utilization rate of multiple modes can be achieved.
(2) This structure can further quantify the expected additional demand to be shifted to yellow taxi rides when a cap is imposed for TNC in the real-world environment, and the city traffic management agency can eventually adapt this approach to achieve fair and dynamic quota regulation during a special holiday or weather conditions, thus distributing traffic volumes within the taxi service zones and reducing the likelihood of traffic accidents.
(3) This methodology could examine potential impact on introducing shared micro-mobility technologies (e.g., e-scooters or e-bikes) within pedestrian infrastructure (Harwood, 2020), proposing a process to integrate other app-based mobility services (i.e., FHV) or medallion taxis with the new mobility, particularly in a local community in order to ensure users' safety and accessibility for the mobility service (Pineda, 2019).

Other medium-term future research tasks include exploring additional factors associated with the demand of yellow taxis or ride-hailing services. Particularly, taxi usage patterns recognized by many other explanatory variables such as socio-demographic or the spatial characteristics should be examined by employing either survey-based or social

media-based travel behaviour data sets. Then, fusing multi-sourced data (Wang et al., 2019) to mitigate the variance of the time series data is required. Furthermore, constructing different types of neural networks to interpret the prediction results is interesting and beneficial. We can also better quantify the contribution of aforementioned features to the forecasting, selecting critical variables (Ribeiro et al., 2016; Lundberg and Lee, 2017). As an example, we could introduce logistic regression or discrete choice models to handle probabilities of utilizing yellow taxis or ride-hailing services by individuals. This interpretable machine learning techniques could lead to an enhanced conceptual modelling structure:

$$\textit{True demand} = \textit{Interpretable trend pattern} + \textit{Structure deviation} + \textit{Seasonal factors} + \textit{Random noise}$$

Therefore, in terms of broader application of the proposed method, we also see other possibilities of quantifying the impact related to new urban mobility policies, e.g., the regulation of fleet sizes dynamically using the frequency of rides per day in a given time and space and the sustainability of deploying a specific number of vehicles within a service area, as well as ensuring the accessibility of emerging technologies (e.g., shared micro-mobility or ride-hailing services) through income-based payment plans (NACTO, 2019). This data driven analytics approach could shed some light on understanding, modeling and systematically managing the relationship between emerging mobility and existing services, and potentially lead to a more optimized vehicle routing and supply system through pre-trip scheduling and on-line dispatching for autonomous vehicles (Shen et al., 2019; Mao et al., 2020).

CHAPTER 3

3. COMPUTATIONAL GRAPH-BASED EFFICIENT COMPUTING FRAMEWORK FOR INTEGRATING ECONOMETRIC MODELS AND MACHINE LEARNING ALGORITHMS

Note: *The substantial part of this chapter is followed by the publication: Kim, T., Zhou, X. and Pendyala, R.M., 2021. Computational Graph-based Framework for Integrating Econometric Models and Machine Learning Algorithms in Emerging Data-Driven Analytical Environments. Transportmetrica A: Transport Science, (just-accepted), pp.1-35.*

In an era of big data and emergence of new technologies such as app-based ride services, there are growing opportunities for better understanding human mobility patterns from newly available data sources. Statistical models have been mainly utilized to uncover and rigorously calibrate the influence of significant factors; and machine learning algorithms have been used to explore complex patterns through improved computing efficiency for large datasets. Focusing on discrete choice modeling applications, this research aims to introduce an open-source computational graph (CG)-based modeling framework for integrating the strengths of econometric models and machine learning algorithms. In particular, multinomial logit (MNL), nested logit (NL), and integrated choice and latent variable (ICLV) models are selected to demonstrate the performance of the proposed graph-oriented functional representation. Furthermore, the calculation of the gradient in the log-likelihood function and associated Hessian matrix is systematically accomplished using automatic differentiation (AD). Using the 2017 National Household Travel Survey data and an open-source dataset, we compare estimation results from the proposed methods with those obtained from two open-source packages, namely Biogeme and Apollo. The results indicate that the CG-based choice modeling approach can produce consistent

estimates of parameters and accurate calculations for the gradients of the estimated parameters with substantial computational efficiency.

3.1. Introduction

The emergence of massive datasets and widespread internet accessibility across the world have offered valuable opportunities for exploring interconnection between physical/cyber infrastructures and human mobility patterns. This has fostered development of techniques to fuse and analyze multiple data sources such as travel surveys, mobile phone data records, GPS, or sensor data (Hashem et al., 2016; Chen et al., 2016; Wu et al., 2018; Chen and Kwan., 2020). With growing interests to explore available data sources, many scholars have executed machine learning methods to efficiently estimate complex hidden patterns in large-scale datasets. In the field of transportation systems, data-driven approaches have been used to identify patterns of diverse traffic flows as well as assist decision makers to predict future trends (Bhavsar et al., 2017; Chang et al., 2019; Zhao et al., 2020). More recently, the research community has taken further steps to develop interpretable machine learning techniques while significant progress has been made in selecting significant variables that affect travel-related choices, enabling the explanation and testing of predicted results (Ribeiro et al., 2016; Lipton, 2018; Molnar, 2020). These research streams point to a potential paradigm shift in transportation demand modeling.

Transportation planners have also recognized that machine learning methods demonstrate high predictive performance and computing efficiency for large-scale mobility datasets, but those data-driven approaches still need to systematically meet standard requirements and expectations associated with modeling travel data sets (e.g., travel

surveys) in transportation planning. The desirable statistics-oriented features include illustrating causal relationships, avoiding overfitted results in relatively small data sets, as well as generating robust standard error estimates for hypothesis testing. If a model estimates only the correlation in a given data set, as pointed out by Mokhtarian (2018), the causation would be eliminated, impeding the ability to answer "why" and "what might happen if" questions. Importantly, incorporating these factors enables researchers and decision makers to deeply fathom the traveler's behavioral patterns. In light of this, statistical modeling approaches have generally been applied in explaining the cause-and-effect relationship and analyzing travel survey data (Paredes et al., 2017; Brathwaite and Walker, 2018b).

In order to bridge the gap between both modeling approaches (i.e., statistical models and machine learning algorithms), this research aims to present a computational framework that can leverage capabilities of existing machine learning platforms to tackle classical estimation problems for discrete choice models. Using a traditional household travel survey dataset and a synthetic dataset available in the Apollo econometric modeling R package, we show how to construct a flexible and efficient modeling framework that utilizes data-driven algorithms in estimating econometric models. The suggested approach could be useful in tackling other estimation problems, such as analyzing multi-dimensional samples from passively collected big data (spatio-temporal dimensions) and enabling real-time updates (predictions) in transportation systems (Nuzzolo and Comi, 2016).

The concept of computational graphs (CGs) is systematically introduced to establish an extended statistical modeling platform capable of covering large-scale datasets

and non-linear architectures (e.g., deep neural networks (DNNs)). The computational graph (CG)-based choice models can take full advantage of automatic differentiation (AD) techniques, which have been widely used in machine learning fields (Abadi et al., 2016; Baydin et al., 2017; Paszke et al., 2017). Three different discrete choice models in transportation planning, namely, multinomial logit (MNL), nested logit (NL), and integrated choice and latent variable (ICLV) functions, are reformulated as computational graphs to estimate parameters and associated statistical properties such as standard errors. These three model forms are chosen because of their widespread use in the field of travel choice modeling. We also examine the flexibility of the modeling structure, and its capability of handling non-concave likelihood functions and simulation-based evaluation of multi-dimensional integrals in latent variable models. Open-source packages, Biogeme (Bierlaire, 2003) and Apollo (Hess and Palma, 2019) are used as test benchmarks, with the publicly accessible National Household Travel Survey (NHTS) 2017 dataset and the synthetic dataset available in the Apollo package serving as use cases.

The remainder of this paper is organized as follows. Section 3.2 presents the literature review with a particular focus on the integration of statistical models and machine learning methods. Section 3.3 describes the National Household Travel Survey (NHTS) 2017 and the synthetic datasets. In section 3.4, the computational graph-based choice models are presented in detail with an emphasis on meeting estimation expectations in planning applications. The estimation and benchmarking results are discussed in section 3.5.

3.2. Literature Review

This section addresses three aspects: integration of discrete choice models and machine learning methods, optimization algorithms, and techniques for computing gradients in objective functions. Focusing on the concept of computational graph (CG) and its example, we also provide a discussion of the motivations behind our proposed approach.

3.2.1. Integration of Choice Models and Machine Learning Algorithms

Recently, research communities have studied hybrid modelling approaches to integrate strengths of machine learning algorithms into discrete choice models (DCMs). For example, Sifringer et al. (2018) proposed a hybrid modeling framework for combining neural networks and multinomial logistic (MNL) models. Selecting the input features that are relatively uncorrelated with choice alternatives, dense neural network (DNN) learned hidden patterns were derived and the trained information was transmitted into the utility function defined in MNL. This methodology interpreted the specified parameters and led to higher log-likelihood values and improved predictive power. Han et al. (2020) further developed an extended framework to integrate MNL and the constrained data-driven structure (multi-layer perceptron (MLP)). Embedding MLP into the utility function of MNL, their approach demonstrated better predictive performance while maintaining the interpretability and preventing the model from over-fitting. More recently, Sifringer et al. (2020) showed the enhanced choice models by embedding neural networks into the specified utility functions of the MNL and NL models. In a residual logit (ResLogit) model proposed by Wong and Farooq (2019), recursive residual layers were constructed in the utility function of the standard MNL model to capture unobserved heterogeneity. Overall,

these above-mentioned modeling efforts aim to resolve overfitting while preserving the econometric interpretability.

Although significant progress has been made to integrate machine learning algorithms in DCM, there are still many challenges to be addressed. First, the existing hybrid models (Sifringer et al., 2018; Han et al., 2020; Sifringer et al., 2020) estimate parameters mainly based on the *Adam* optimizer proposed by Kingma and Ba (2014) or stochastic gradient descent (SGD) (Bottou, 2010). In terms of optimizing objective functions, the first order-based estimators can be computationally effective to analyze a large-scale dataset and calibrate numerous parameters. However, we have to recognize that there are various model structures in which we are dealing with non-concave functions (e.g., nested logit (NL) model (Williams, 1977)) or simulation-based models involving computation of high-dimensional integrals such as the integrated choice and latent variable (ICLV) model (Ben-Akiva et al., 2002) and the hybrid choice model with a nonlinear utility function (Kim et al., 2016). Second, the first order-based estimation might not be able to provide desirable statistical properties in computing the Hessian matrix. These challenges require a systematic and careful analysis for an effective combination of machine learning techniques and optimization algorithms in the context of statistically-oriented choice models for transportation applications.

3.2.2. Optimization Algorithms for Discrete Choice Models

In the area of discrete choice modeling, maximum likelihood estimation (MLE) is one of the fundamentally important estimation methods. By computing the first order (gradient) and second order (curvature) derivatives of the likelihood function, MLE furnishes values

of parameters by maximizing the likelihood function through the use of the Hessian matrix. The derivatives are computed by three approaches: manual/analytical, finite difference, and automatic differentiation (AD) (Bartholomew et al., 2000). Due to the difficulty of embedding/coding highly nonlinear forms in complicated functions, manual differentiation could be used for some very small cases. The numerical differentiation aims to approximate derivatives through the finite differencing, but the solution quality is greatly affected by the potential truncation and round-off errors associated with different finite difference formulas (Wright and Nocedal, 1999). On the other hand, the automatic differentiation (AD) technique utilizes the chain rule-based principle and intermediate variables to evaluate complex derivatives analytically (Wright and Nocedal, 1999; Griewank, and Walther, 2008). Specifically, in the new generation of low-level computational graph libraries such as Tensorflow and PyTorch, the computing architecture can enable modelers to represent the analytical optimization model through a graph of simple elementary operations (i.e., addition, subtraction, multiplication, and division) and elementary functions (e.g., natural logarithm), and further execute a sequential and complex structure of computations easily. In new domain-specific languages (DSLs) for convex optimization such as CVXPY, progress has been made recently to convert standard convex optimization to detailed CG representations with low-level solver interfaces (Agrawal et al. 2018). It should be noted that AD might still encounter the difficulty of computing piecewise rational functions, especially when estimating gradients of non-smooth composite functions (Beck and Fischer, 1994; Nocedal and Wright, 2006).

In the machine learning area, the sequential structure and computational graph approach have been widely applied for large-scale datasets with numerous parameters to be calibrated. These applications have demonstrated the capability of these approaches in computing gradients and Hessians of non-linear optimization formulations efficiently and precisely (Baydin et al., 2017). From a specific system identification perspective, the AD technique has been utilized in the fields of machine learning and econometric modeling to estimate parameters, thanks to its computational efficiency and flexibility of designing diverse composite functions (Sifringer et al., 2018; Wong and Farooq, 2019; Sun et al., 2019; van Kesteren and Oberski, 2019; Han et al., 2020). Furthermore, in the case of discrete choice modeling (DCM), by carefully selecting the underlying computing algorithms, AD holds the promise for more precise computation of derivatives of the log likelihood with respect to specified parameters through chain rules and back propagation. That is, simply using the popular first order methods (e.g., SGD or *Adam*) is often inadequate in estimating complicated modeling structures (e.g., NL or ICLV). Thus, our research combines the AD technique with quasi-second order methods, e.g., Broyden-Fletcher-Goldfarb-Shanno (BFGS), to calibrate non-concave composite functions and deliver consistent statistical estimates through Hessians.

3.2.3. Computational Graph (CG)

Understanding computational graph (CG) approach is important for designing flexible modeling structures that integrate choice models and machine learning seamlessly. Using the binary logit model in Eq. (1) as an example, Wu et al., (2018) and Sun et al. (2019)

took a few initial steps to illustrate how CG can decompose complex composite functions as follows.

$$\mathrm{P}\,(y = 1) = \frac{1}{1 + e^{-V}} \tag{3.1}$$

Eq. (1) indicates the probability of choosing a binary alternative, and the term V is a specified utility function (e.g., $V = \beta_0 + \beta_1 x_1 + \beta_2 x_2 + \cdots + \beta_n x_n$ where β_n is the unknown parameter associated with the attribute x_n). Using the concept of computational graph (CG), this logistic function is now expressed as a directed graph which consists of nodes (elementary operations) and edges (directions):

Figure 3.1 clearly illustrates the logistic formulation written in Eq. (3.1) as a sequentially nested structure made up of nodes and edges. In particular, Figure 3.1(a) is the process of computing the probability of a given binary alternative, and Figure 3.1(b) represents the procedure of estimating parameters. For example, the parameter β_1 is obtained by the defined nodes and links shown in Figure 3.1:

Eq. (3.2) presents the analytic derivative with respect to the parameter and the description of the chain rule-based computation. Furthermore, applying the gradients in the BFGS optimizer, this computed differentiation offers more precise Hessians. In this context, it is helpful to compare the computed values in Eq. (3.2) with analytical sensitivities detailed in Koppelman and Bhat (2006) and Train (2009).

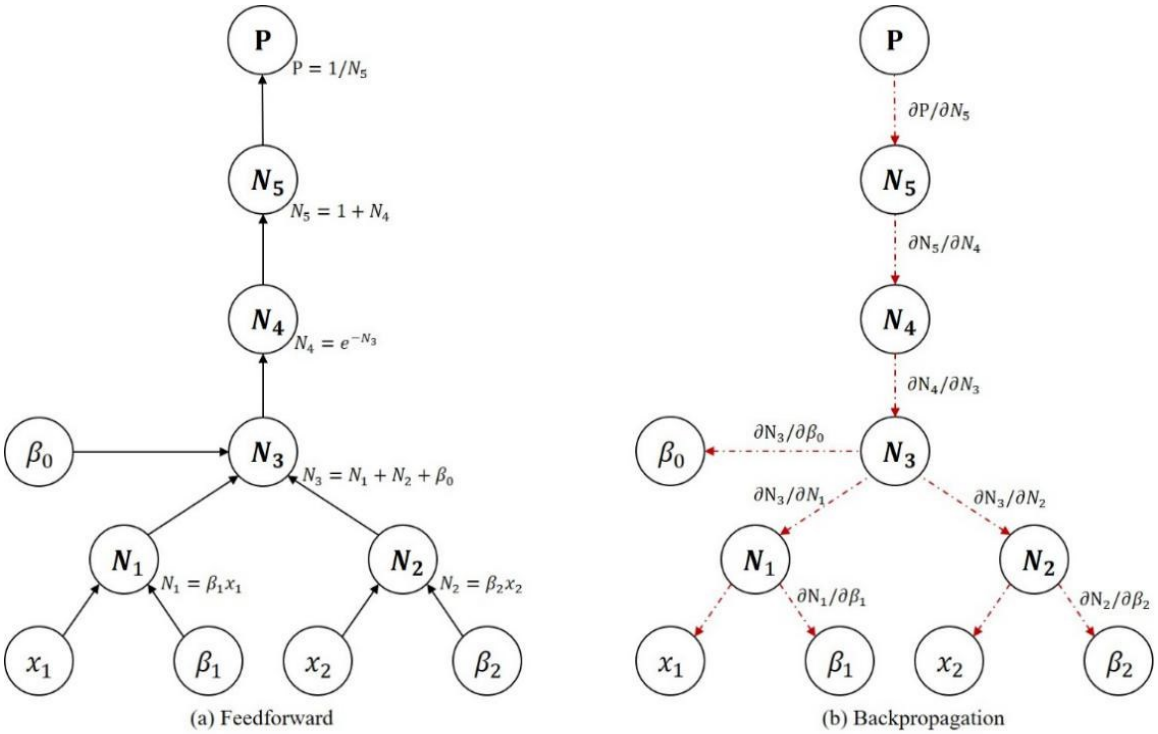


Figure 3.1. Computational Graph (CG) of the Binary Logit Model

$$\frac{\partial \mathrm{P}}{\partial \beta_1} = \frac{\partial \mathrm{P}}{\partial N_5} \cdot \frac{\partial N_5}{\partial N_4} \cdot \frac{\partial N_4}{\partial N_3} \cdot \frac{\partial N_3}{\partial N_1} \cdot \frac{\partial N_1}{\partial \beta_1} = \frac{x_1}{(N_5)^2} \cdot \mathrm{e}^{-N_3} = \frac{x_1}{(1+e^{-V})^2} \cdot \mathrm{e}^{-V} \qquad (3.2)$$

To calibrate a broader set of DCMs in transportation planning with rigorously defined standard error estimates, we will tackle three econometric models (i.e., multinomial logit, nested logit , and integrated choice and latent variable) to demonstrate the capability of the enhanced choice modelling framework along three directions: the numerical efficiency of processing a high-dimension survey sample, greater flexibility in employing different composite functions (e.g., deep learning architectures), and realization of desirable statistical properties. A widely used machine learning platform, TensorFlow (Abadi et al., 2016), is selected to implement the proposed CG-based discrete choice models, and the source code can be downloaded at Kim et al. (2021). There are other computational graph-oriented programming platforms such as Theano (Bastien et al., 2012)

or Pytorch (Paszke et al., 2017). In addition, to systematically verify the estimated parameters and statistical properties, two leading open-source packages for estimating DCMs, namely Biogeme (Bierlaire, 2003) and Apollo (Hess and Palma, 2019), are used to serve as benchmarks.

It should be noted that the concept of computational graph has been adapted in the pioneering open-source DCM estimation package, Biogeme, in 2000, through the use of chain rule differentiation and analytical gradients. In our proposed domain-specific languages (DSLs) for maximum likelihood estimation of various DCMs, we do not need to build the low-level computational graph manually through a general-purpose language (GPL); instead, we translate the corresponding DCM optimization to forms compatible to the interfaces of recent CG libraries (e.g., TensorFlow). By doing so, our approach can further fully utilize the backpropagation mechanism provided by differentiable optimization layers/pipelines. The DSLs for MLE-DCM helps modelers greatly reduce the computational redundancy by decomposing the computing units in a layered structure and enabling the use of dynamic programming for iteratively finding a solution. The development of domain-specific languages requires a deep understanding of the problem structure and domain knowledge, and we will further highlight the potential for integrating different transportation modeling elements of more complex estimation and planning problems in the conclusion of this paper.

3.3. Data Preparation

Two datasets are utilized in this research: the 2017 National Household Travel Survey (NHTS) dataset for estimating MNL and NL models, and a synthetic dataset provided by

Hess and Palma (2019) to estimate the extended integrated choice and latent variable or ICLV model.

3.3.1. National Household Travel Survey (NHTS) Dataset

The dataset used for the case study is derived from the National Household Travel Survey (NHTS 2017) conducted by the US Department of Transportation. This data set provides information about travel behavior, particularly associated with trip purposes and modes. In the current study, this large-scale dataset with 923,572 trips is explored. To alleviate unobserved taste heterogeneity, we restrict the scope of the trip purpose and time-dimension by selecting commuting trips (home to work trips) departing between 6 and 9 AM.

After filtering the dataset based on criteria and eliminating obviously erroneous observations or those with large amounts of missing data, the final subsample size used for the model estimation is 40,177 observations. Table 3.1 depicts the travelers' socio-economic and demographic information, as well as travel time and distance variables that are subsequently used as explanatory variables in the specification of the utility function. The five alternatives, namely drive alone (DA), shared ride (SR), transit (TR), bike, and walk, are considered as the choice elements in the proposed MNL and NL choice models. In terms of person characteristics, 84.3 percent of the commuting trips are accounted for by those age 30-74 years. The gender ratio of this subsample is nearly 51 percent male and 49 percent female. In terms of educational attainment, travelers who earned the bachelor's degree and graduate degree account for 29.8 percent and 26.1 percent of the commute tours, respectively. Among household attributes, individuals within the household income

categories ($50,000-$124,999 and $125,000 or above) account for 76.7 percent of the commute tours. Two-person households and individuals living with five persons or more account for the highest and lowest proportion of commute tours, respectively. Nearly 79 percent of commuters travel from an urban area. According to travel characteristics, the average commute distance is 12.9 miles with a standard deviation of 15.8 miles, and the average time taken is 27.3 minutes with a standard deviation of 27.9 minutes. The distribution of commute mode choices is 79.3 percent of commute trips by drive alone (DA), 13.8 percent by shared ride, 3.8 percent by transit, and 3.1 percent by bike and walk. This mode choice distribution follows a similar pattern in a prior study by Paleti et al. (2013).

Table 3.1. Description of the subsample (N=40,177)

Person characteristics	**Frequency**	**Percentage (%)**
Age		
Less than 18 years	111	0.3
18-24 years	2,259	5.6
25-29 years	3,549	8.8
30-44 years	11,502	28.6
45-59 years	15,094	37.6
60-74 years	7,270	18.1
75 years or above	392	1.0
Gender		
Male	20,387	50.7
Female	19,790	49.3
Education attainment		
Less than bachelor's degree	17,723	44.1
Bachelor's degree	11,976	29.8
Graduate degree	10,478	26.1
Household characteristics	**Frequency**	**Percentage (%)**
Household income		
Under $25,000	2,812	7.0
$25,000 - $49,999	6,560	16.3
$50,000 - $124,999	10,491	26.1
$125,000 or above	20,314	50.6
Household size		

1 (I am the only person)	6,284	15.6
2 people	17,468	43.5
3 people	7,270	18.1
4 people	6,032	15.0
5 people or more	3,123	7.8
Travel characteristics	**Continuous (average)**	
Trip distance in miles & Trip duration in minutes	12.88 miles & 27.33 minutes	
Endogenous variable	**Frequency**	**Percentage (%)**
Trip mode		
Drive alone (DA)	31,872	79.3
Shared ride (SR)	5,530	13.8
Transit (TR)	1,543	3.8
Bicycle	386	1.0
Walk	846	2.1

3.3.2. Synthetic Dataset

The lack of attitudinal questions in the NHTS dataset renders it unsuitable for constructing ICLV components, i.e., structural models with latent variables and measurement equations. As a result, we utilized an alternative synthetic dataset that accompanies the Apollo package to estimate the ICLV model (instead of using the NHTS dataset). This dataset documents drug choices for 1,000 individuals; four alternative choices, three socio-demographic characteristics, and four attitudinal questions are presented. The explanatory variables to construct the structural equation of a latent variable were binary in nature: regular drug users, university degree attainment, and age 50 years and above. In addition, the attitudinal questions to define the measurement equations followed a Likert scale from 1 (strongly disagree) to 5 (strongly agree). Four attitudinal questions are selected as measurement equation indicators. The detailed description of the drug choice data is well documented in Hess and Palma (2019).

3.4. Modeling Framework and Methodology

This section presents the mathematical formulations of MNL, NL, and ICLV models, the computational graph-based modeling frameworks, as well as the stepwise procedure of estimating the proposed graph-oriented functions. Using the travel survey dataset, we develop the systematic utility function and the probability of choice alternatives, namely drive alone (DA), shared ride (SR), transit (TR), bike, and walk, to estimate MNL and NL models. On the other hand, the ICLV components (i.e., the structural equation of the latent variable, measurement indicators, utility functions, as well as the probability of a drug choice between four alternatives) are constructed using the synthetic dataset.

3.4.1. Mathematical Formulations of the MNL and NL Models

With the fundamental assumptions that error components in the utility function are independently and identically distributed according to a Gumbel distribution, the functional formulation of the multinomial logit (MNL) model can be defined clearly. The probability that a decision maker n chooses an alternative mode i among a set of J alternatives (i.e., DA, SR, TR, bike, and walk) is as follows (McFadden, 1974):

$$\mathbf{P}_{n,i} = \frac{e^{V_{n,i}}}{\sum_{j \epsilon J} e^{V_{n,j}}} \tag{3.3}$$

where $V_{n,i}$ denotes the systematic utility of the alternative mode $i \epsilon J$ selected by the decision maker n, and the structural utility function includes alternative specific constants and observed attributes with their parameters (i.e., $V_{n,i} = ASC_{i,n} + \sum_{k=1}^{K} \beta_{k,i} x_{k,i,n}$). The index J is the set of the specified alternative choices. K represents the number of attributes used as choice predictors.

By reformulating the MNL structure to relax the independence of irrelevant alternatives (IIA) property of MNL, the nested logit (NL) can be specified (Williams 1977; McFadden 1978). In particular, two layered structures are considered in this study. The upper level of NL includes drive alone (DA), shared ride (SR), transit (TR), and the non-motorized group, and the two alternatives (i.e., bike and walk) included in the non-motorized group are located in the lower level.

The functional formula of the choice probability is expressed by the product of the conditional probability and the marginal probability. For instance, the probability that a decision maker n selects an alternative i in the nest m is formulated as:

$$\mathbf{P}_{n,i} = \mathbf{P}_{n,i|J_m} \times \mathbf{P}_{n,J_m} = \frac{e^{V_{n,i}/\lambda_m}}{\sum_{l \in J_m} e^{V_{n,l}/\lambda_m}} \times \frac{e^{(V_{n,m}+\lambda_m \Gamma_{n,m})}}{\sum_{j=1}^{M} e^{(V_{n,j}+\lambda_j \Gamma_{n,j})}} \tag{3.4}$$

In Eq. (3.4) the first component is the conditional probability that the decision maker n chooses either a bike or walk mode given that the non-motorized group J_m is selected, and the second component is the marginal probability of choosing between drive alone, shared ride, transit, and the nested group. λ_m is the logsum parameter bounded by zero to one, an indicator of the correlation between bike and walk; the parameter is explained well in Koppelman and Bhat (2006). The inclusive value $\Gamma_{n,m}$ (or often called log-sum term) is defined by $\Gamma_{n,m} = \log\left[\sum_{l \in J_m} e^{V_{n,l}/\lambda_m}\right]$ where this term is associated with the nested group. Readers interested in the derivation of the mathematical formulations can find details in Koppelman and Bhat (2006) and Train (2009).

3.4.2. Mathematical Formulations of the ICLV Model

ICLV incorporates a latent variable model into a multinomial discrete choice model. To enable this integrated model, four components are generally required to be specified; a latent variable, measurement indicators, utility functions, and choice probabilities (Ben-Akiva et al., 2002). First, the latent variable formulated as a function of observable explanatory variables with a stochastic component is given by:

$$X_n^* = \gamma z_n + \eta_n \tag{5}$$

Eq. (3.5) indicates the structural equation for the latent variable X^* influenced by explanatory variables z_n including three socio-demographic characteristics (in this study) with parameters γ. The stochastic term η_n follows a standard normal distribution $\eta_n \sim N(0,1)$. Second, the probability distribution function of the continuous measurement indicators is expressed as follows:

$$f_n(I_n|z_n, X_n^*; \boldsymbol{\zeta}, \boldsymbol{\sigma}) = \frac{1}{\sqrt{2\pi\sigma_k^2}} e^{-\frac{\left(I_{n,k}-\bar{I}_k-\zeta_k X_n^*\right)^2}{2\sigma_k^2}} \tag{3.6}$$

where the continuous measurement indicators are defined by $I_{n,k} = \zeta_k X_n^* + v_n$. $I_{n,k}$ represents an indicator associated with an attitude $k \in K$ and the continuous measurement model. $\bar{I}_k$ is the average of the indicator k. Subtracting it from $I_{n,k}$, we avoid estimating the mean of the normal density. ζ_k is the attitudinal coefficient for the latent variable X_n^*, and v_n is the stochastic component characterized by a standard normal distribution $v_n \sim N(0,1)$. Third, the systematic utility function is specified by $V_{n,i} = \sum_{s=1}^{S} \beta_{s,i} x_{s,n,i} + \lambda X_n^*$, where $\beta_{s,i}$ and λ are coefficients of choice predictors and the latent variable, respectively. $V_{n,i}$ represents the utility function of the alternative drug i selected by the

decision maker n. Lastly, the probability that a decision maker n chooses a drug i among a set of four products is defined by the multinomial logit formulation. Using the defined components above, we can obtain the joint choice probability as follows (Ben-Akiva et al., 2002; Vij and Walker, 2016):

$$\mathbf{P}_i = \int_{\eta_n} \prod_{k=1}^{K} \frac{1}{\sqrt{2\pi\sigma_k^2}} e^{-\frac{\left(I_{n,k}-\bar{I}_k-\zeta_k X_n^*\right)^2}{2\sigma_k^2}} \times \frac{e^{V_{n,i}}}{\sum_{j\epsilon J} e^{V_{n,j}}} \times \phi(\eta_n) d\eta_n \tag{3.7}$$

In Eq. (3.7), the first component is the likelihood of the continuous measurement indicators, the second term is the multinomial logit model, and the third term is derived from the structural equation of the latent variable. Since Eq. (3.7) has no closed-form solution, this joint choice probability function is conventionally approximated using a Monte Carlo simulation-based approach:

$$\mathbf{P}_i \cong \frac{1}{T}\sum_{t=1}^{T} \prod_{k=1}^{K} \frac{1}{\sqrt{2\pi\sigma_k^2}} e^{-\frac{\left(I_{n,k}-\bar{I}_k-\zeta_k X_{n,t}^*\right)^2}{2\sigma_k^2}} \times \frac{e^{V_{n,i,t}}}{\sum_{j\epsilon J} e^{V_{n,j,t}}} \tag{3.8}$$

Drawing the standard normal distribution function η_n iteratively, we can simulate the multidimensional integrals, thus deriving Eq. (3.8); T is the total number of draws. The detailed description of simulation-based approaches can be found in Train (2009). With the above-derived functions, we now present the procedure of constructing computational graph-based models.

3.4.3. Illustration of the Computational Graph-Based Modeling Approach

This subsection presents the CG-based modeling structures for MNL, NL, and ICLV. We present an illustrative example to demonstrate the sequential process of formulating the probability functions associated with mode choices and drug choices in the two datasets

respectively. In this description, the probability of choosing the walk mode is exemplified using MNL and NL, and the probability of selecting a drug between four alternatives is illustrated for the ICLV.

3.4.3.1. CG-based Multinomial Logit Model

Eq. (3.3) is decomposed and plotted into the directed graph, which includes elementary operations and elementary functions. As shown in Figure 3.2, there are 15 input nodes and 16 intermediate nodes to link between input nodes and the output node; input nodes are comprised of the alternative specific constants (ASC) for each alternative and unknown parameters β associated with the attributes x, and the intermediate nodes (N_i where $i = 1, 2, \dots, 16$) play a role of decomposing functions. The output node is the probability of selecting the walk mode $\mathbf{P}_{walk}$. Based on the nodes interconnected by directed edges, we can produce the sequentially nested structure for the probability function so that Eq. (3.3) can be mapped as follows:

$$\begin{aligned} \mathbf{P}_{walk} &= N_{15}/N_{16} \\ &= \mathrm{e}^{N_{10}}/(N_{11} + N_{12} + N_{13} + N_{14} + N_{15}) \\ &= e^{(N_5 + ASC_{walk})}/(e^{N_6} + e^{N_7} + e^{N_8} + e^{N_9} + e^{N_{10}}) \\ &= e^{(\beta_{walk} x_{walk} + ASC_{walk})}/\big(e^{(N_1 + ASC_{DA})} + e^{(N_2 + ASC_{SR})} + e^{(N_3 + ASC_{TR})} + \end{aligned} \tag{3.9}$$

where the nodes from N_{16} to N_5 are used to connect input nodes and the output node, and the index i represents the labels of choice alternatives (DA, SR, TR, Bike, and Walk). It should be noted that in order to simplify the illustration, nodes associated with the availability of the given alternatives are excluded in this graph.

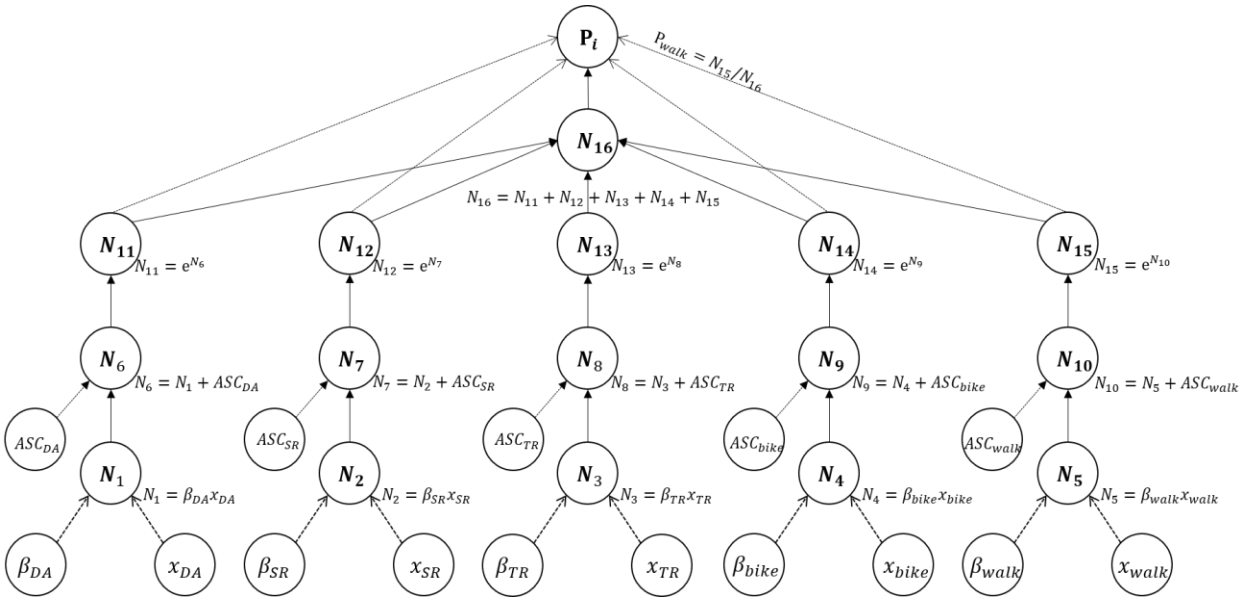


Figure 3.2. Illustration of CG-Based Multinomial Logit Model

3.4.3.2 CG-Based Nested Logit Model

A two-level nested structure is described in this subsection. Based on Eq. (3.4), the probability of selecting the walk mode is plotted in Figure 3.3. In contrast to the MNL model, this nested model is formulated using the conditional probability and marginal probability to account for the correlation between bike and walk. Figure 3.3 denotes 21 input nodes including the nodes used in the MNL computational graph, the log-sum parameter λ_m, as well as the log-sum function Γ_{nm}. In addition, 27 intermediate nodes are embedded to express the decomposed components of NL. With the specified nodes and the directed edges, the product of the conditional probability and the marginal probability can be computed to derive the probability of selecting the walk mode $\mathbf{P}_{walk}$ as follows:

$$\mathbf{P}_{walk} = \mathbf{P}_{walk|non-auto}\mathbf{P}_{non-auto} = N_{26}N_{27} \tag{3.10}$$

The conditional probability $\mathbf{P}_{walk|non-auto}$ is equal to N_{26}, and the term N_{27} indicates the marginal probability of falling into the non-auto group. To be specific, the sequential steps of mapping the conditional probability $\mathbf{P}_{walk|non-auto}$ are detailed below:

$$\begin{aligned}
\mathbf{P}_{walk|non-auto} &= N_{20}/N_{22} \\
&= e^{N_{15}}/(N_{19}+N_{20}) \\
&= e^{N_9/\lambda_m}/(e^{N_{14}}+e^{N_{15}}) \\
&= e^{(N_4+ASC_{walk})/\lambda_m}/(e^{N_9/\lambda_m}+e^{N_{10}/\lambda_m}) \\
&= e^{(\beta_{walk}x_{walk})/\lambda_m}/(e^{\beta_{bike}x_{bike}/\lambda_m}+e^{\beta_{walk}x_{walk}/\lambda_m})
\end{aligned} \tag{3.11}$$

Eq. (3.11) illustrates a stepwise procedure for deriving the conditional probability. The detailed description of the CG nodes and links can be found in Figure 3.3. Similarly, the marginal probability $\mathbf{P}_{non-auto}$,which is mapped by the forward propagation of the CG framework, can be written in the following stepwise manner:

$$\begin{aligned}
\mathbf{P}_{non-auto} &= N_{24}/N_{25} \\
&= \lambda_m N_{23}/(N_{21}+N_{24}) \\
&= \lambda_m \log(N_{22})/(N_{16}+N_{17}+N_{18}+\lambda_m N_{23}) \\
&= \lambda_m \log(N_{19}+N_{20})/(e^{N_{11}}+e^{N_{12}}+e^{N_{13}}+\lambda_m N_{23}) \\
&\vdots \\
&= \lambda_m \log\left(e^{\frac{N_9}{\lambda_m}}+e^{\frac{N_{10}}{\lambda_m}}\right)\Big/\left(e^{\frac{N_6}{\lambda_m}}+e^{\frac{N_7}{\lambda_m}}+e^{\frac{N_8}{\lambda_m}}+\lambda_m \log\left(e^{\frac{N_9}{\lambda_m}}+e\right.\right. \\
&= \frac{\lambda_m \log\left(e^{(\beta_{bike}x_{bike})/\lambda_m}+e^{(\beta_{walk}x_{walk})/\lambda_m}\right)}{(e^{N_6/\lambda_m}+e^{N_7/\lambda_m}+e^{N_8/\lambda_m}+\lambda_m log(e^{N_9/\lambda_m}+e^{N_{10}/\lambda_m}))}
\end{aligned} \tag{3.12}$$

Eq. (3.12) is the marginal probability of falling into the non-auto nest. The expression $\log\left(e^{(\beta_{bike}x_{bike})/\lambda_m}+e^{(\beta_{walk}x_{walk})/\lambda_m}\right)$ corresponds to the log-sum function Γ_{nm}. By computing the product of Eq. (3.11) and (3.12), we can now derive the probability function Eq. (3.10) through the graph-oriented function. Please note that the utility function V_{nm} shown in Eq. (3.4) is assumed as zero.

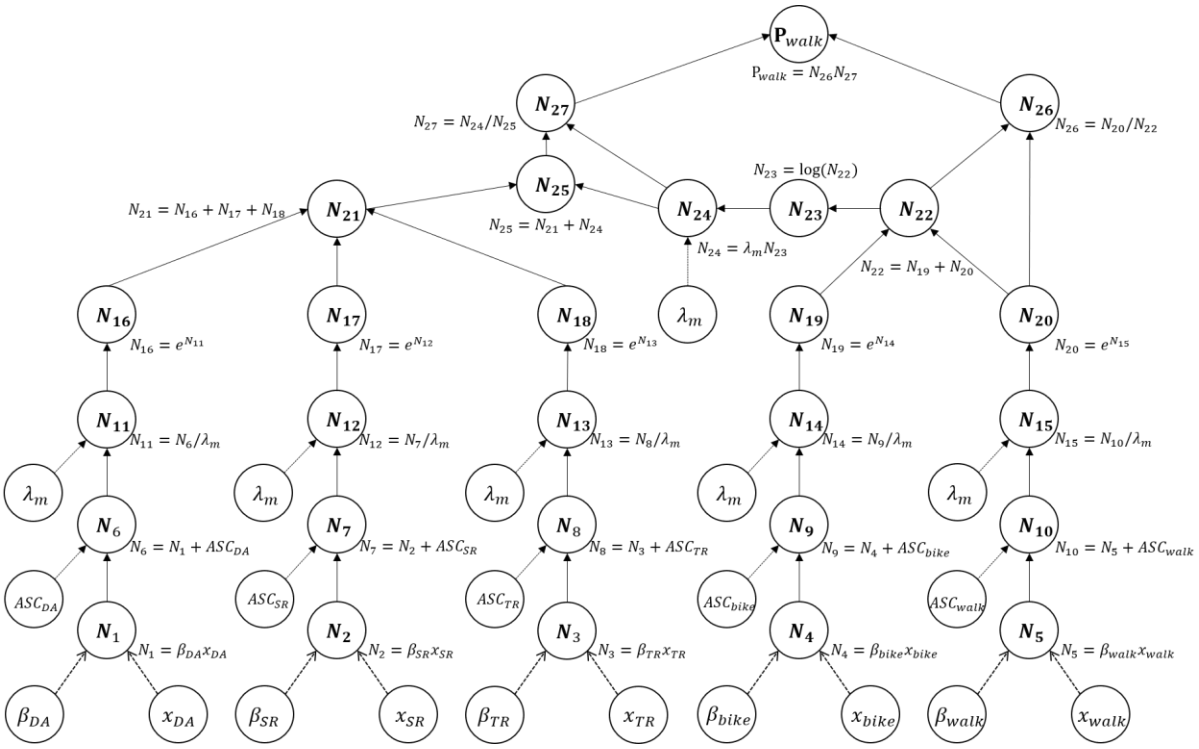


Figure 3.3. Illustration of CG-Based Two-Level Nested Logit Model

3.4.3.3 CG-Based Integrated Choice and Latent Variable (ICLV) Model

In this subsection, the ICLV function comprising of one latent variable, the stochastic term, continuous measurement indicators, as well as the multinomial logit structure is decomposed and plotted in a series of nodes (elementary operations) and edges (directions). According to Figure 3.4, 17 input nodes and 22 intermediate nodes are used. N_6, N_{14}, N_{22}, and N_{23} are used to denote the ICLV components, in order to develop the output node $\mathbf{P}_{A1}$ which is the joint choice probability of choosing a drug between four alternatives. With the CG-based structure, the exemplified choice probability can be written in the stepwise manner:

$$\begin{aligned} \mathbf{P}_{A1} &= N_{22} \times N_{23} \\ &= (N_{21}/N_{18}) \times (N_{10}/N_{14}) \end{aligned} \tag{3.13}$$

$$
\begin{aligned}
&= e^{-N_{20}}/\sigma\sqrt{2\pi} \times e^{N_8}/(N_{10} + N_{11} + N_{12} + N_{13}) \\
&\vdots \\
&= \frac{1}{\sigma\sqrt{2\pi}} e^{-\frac{(I-\zeta N_6)^2}{2\sigma^2}} \\
&\quad \times \frac{e^{(N_1+\lambda N_6)}}{(e^{(N_1+\lambda N_6)} + e^{(N_2+\lambda N_6)} + e^{N_3} + e^{N_4})}
\end{aligned}
$$

Eq. (3.13) denotes the joint choice probability of falling into drug alternative 1. The first component corresponds to the measurement indicators, while N_6 is the structural equation of the latent variable. The second term is the discrete choice formulation. In order to simplify the illustration shown in Figure 3.4, we only show the first iteration of the simulated choice model and exclude nodes associated with the availability of the given alternatives.

With the underlying knowledge of building the forward propagation of the CG-based choice models, the following subsection discusses the automatic differentiation (AD) algorithm to estimate the proposed CG-based choice models in a backpropagation approach. We describe the backpropagation step by step using the plotted figures.

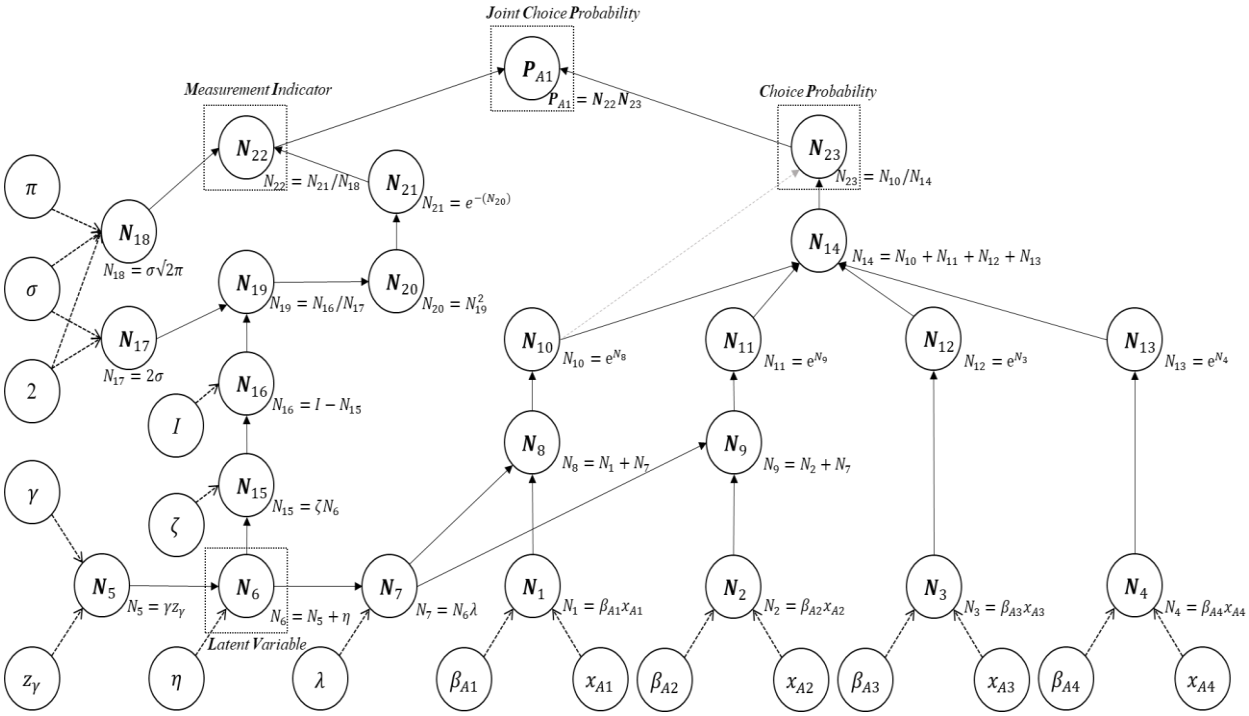


Figure 3.4. Illustration of CG-Based Integrated Choice and Latent Variable (ICLV) Model

3.4.4. Parameter Estimation: Automatic Differentiation (AD) with BFGS

In the CG-based architecture, the unknown parameters specified in Eq. (3), (4), and (8) can be estimated by minimizing the negative log-likelihood function, and the corresponding objective function leads to a particular type of the categorical cross-entropy function proposed by Shannon (1948).

$$H_n(\mathbf{P}_n, \boldsymbol{y}_n) = -\sum_{i \in J} y_{n,i} \ln(\mathrm{P}_{n,i}(\beta)) \tag{3.14}$$

where $\mathrm{y}_{n,i}$ is the discrete variable that denotes a choice $i \in J$ selected by a decision maker n. Eq. (3.14) is commonly expressed as $LL(\beta)$, log-likelihood, in the discrete choice field. Using the second-order Taylor's approximation of log-likelihood function $LL(\beta_{k+1})$ in a

neighborhood of $LL(\beta_k)$, we can find the optimal value of parameters β_{k+1} to maximize $LL(\beta_{k+1})$ (Train, 2009).

$$\frac{\partial LL(\beta_{k+1})}{\partial \beta_{k+1}} = \frac{\partial LL(\beta_k)}{\partial \beta_k} + B_k(\beta_{t+1} - \beta_t) = 0 \tag{3.15}$$

The partial derivative of $LL(\beta_k)$ with respect to β_k and the numerically approximated Hessian matrix B_k are determining the best value of β_{k+1}. More specifically, when solving Eq. (3.15), β_{k+1} can be expressed as $\beta_k + (-B_k)^{-1}(\partial LL(\beta_k)/\partial \beta_k)$. In order to compute the first-order gradients of the objective function with respect to each parameter, we utilize the automatic differentiation (AD) algorithm. By utilizing the derived gradients in the BFGS optimizer, we can calculate the Hessian matrix which is used to evaluate statistical properties of estimated parameters. A detailed description of computing the numerical Hessian matrix is explained in Nocedal and Wright (2006). As illustrated in the study, the first-order gradient information is valuable for assisting the chain rule-based algorithmic differentiation procedure in deriving the gradients in each choice model.

Consider the estimation of the parameter β_{walk} shown in the equations. The numerical derivative of the parameter in MNL can be derived by the chain rule.

$$\begin{aligned}
\frac{\partial LL(\beta_{walk})}{\partial \beta_{walk}} &= \frac{\partial LL(\beta_{walk})}{\partial \mathrm{P}_{walk}} \frac{\partial \mathrm{P}_{walk}}{\partial N_{16}} \frac{\partial N_{16}}{\partial N_{15}} \frac{\partial N_{15}}{\partial N_{10}} \frac{\partial N_{10}}{\partial N_5} \frac{\partial N_5}{\partial \beta_{walk}} \\
&= \frac{1}{\mathrm{P}_{walk}} \frac{(N_{16} - N_{15})}{(N_{16})^2} e^{N_{10}} \boldsymbol{x}_{walk} \\
&= \frac{1}{\mathrm{P}_{walk}} \frac{(e^{N_6} + e^{N_7} + e^{N_8} + e^{N_9})}{(e^{N_6} + e^{N_7} + e^{N_8} + e^{N_9} + e^{N_{10}})^2} e^{N_{10}} \boldsymbol{x}_{walk}
\end{aligned} \tag{3.16}$$

Eq. (3.16) further details the sequential procedure of computing the partial derivative of $\mathbf{P}_{walk}$ defined in Eq. (3.9) with respect to the parameter β_{walk}. The

description of the intermediate nodes (N_i where i = 6, 7, 8, 9, 10) is illustrated in Figure 3.2. The rest of the parameters defined in the CG-based MNL model can be calculated similarly. Now, utilizing the computational graph for the NL model, we introduce the stepwise procedure for computing the partial derivative of the log-likelihood of $\mathbf{P}_{walk}$ with respect to the parameter β_{walk} in Eq. (3.17).

$$\begin{aligned}
\frac{\partial LL(\beta_{walk})}{\partial \beta_{walk}} &= \frac{\partial LL(\beta_{walk})}{\partial \mathrm{P}_{walk}} \frac{\partial \mathrm{P}_{walk}}{\partial N_{26}} \frac{\partial N_{26}}{\partial N_{20}} \frac{\partial N_{20}}{\partial N_{15}} \frac{\partial N_{15}}{\partial N_{10}} \frac{\partial N_{10}}{\partial N_5} \frac{\partial N_5}{\partial \beta_{walk}} \\
&= \frac{1}{\mathrm{P}_{walk}} \frac{N_{27}}{N_{22}} e^{N_{15}} \frac{1}{\lambda_m} \boldsymbol{x}_{walk} \\
&= \frac{1}{\mathrm{P}_{walk}} \frac{\lambda_m N_{23}}{(N_{16} + N_{17} + N_{18} + \lambda_m N_{23})} \frac{e^{N_{15}}}{\lambda_m} \boldsymbol{x}_{walk} \\
&= \frac{1}{\mathrm{P}_{walk}} \frac{\log(e^{N_{14}} + e^{N_{15}})}{(e^{N_{11}} + e^{N_{12}} + e^{N_{13}} + \lambda_m \log(e^{N_{14}} + e^{N_{15}}))} e^{N_{15}} \boldsymbol{x}_{walk}
\end{aligned} \tag{3.17}$$

In the nesting structure, we can observe the log-sum parameter λ_m and inclusive value term as $\log(e^{N_{14}} + e^{N_{15}})$, and the probability of P_{walk} is as shown in Eq. (3.10). In a similar manner, the stepwise procedure of estimating the partial derivative of the log-likelihood of P_{A1}, Eq. (3.13), with respect to the parameter β_{A1} in the ICLV model can be expressed as:

$$\begin{aligned}
\frac{\partial LL(\beta_{A1})}{\partial \beta_{A1}} &= \frac{\partial LL(\mathrm{P}_{A1})}{\partial \mathrm{P}_{A1}} \frac{\partial \mathrm{P}_{A1}}{\partial N_{23}} \frac{\partial N_{23}}{\partial N_{14}} \frac{\partial N_{14}}{\partial N_{10}} \frac{\partial N_{10}}{\partial N_8} \frac{\partial N_8}{\partial N_1} \frac{\partial N_1}{\partial \beta_{A1}} \\
&= \frac{1}{\mathrm{P}_{A1}} N_{22} \left(- \frac{N_{10}}{(N_{14})^2} \right) e^{N_8} \boldsymbol{x}_{A1} \\
&= \frac{1}{\mathrm{P}_{A1}} N_{22} \left(- \frac{e^{N_8}}{(e^{N_8} + e^{N_9} + e^{N_3} + e^{N_4})^2} \right) e^{N_8} \boldsymbol{x}_{A1} \\
&= \frac{1}{\mathrm{P}_{A1}} \frac{1}{\sigma\sqrt{2\pi}} e^{-\frac{(I - \zeta N_6)^2}{2\sigma^2}} \left(- \frac{e^{N_8}}{(e^{N_8} + e^{N_9} + e^{N_3} + e^{N_4})^2} \right) e^{N_8} \boldsymbol{x}_{A1}
\end{aligned} \tag{3.18}$$

With the computed gradients of the log-likelihood function, the TensorFlow-based program starts from the initial settings of parameters and convergence criteria. Then these numerical tensors are transmitted into the optimizer of BFGS relying on an approximated Hessian matrix, with the goal of minimizing the negative log-likelihood function defined by the CG-based structure. Based on the iterative algorithm of the optimizer, the inverse of the Hessian matrix $\widehat{H}^{-1}$ is derived such that we can obtain the parameter variance-covariance matrix as follows:

$$\begin{aligned} \mathrm{SE}\left(\widehat{\boldsymbol{\beta}}\right) &= \sqrt{\frac{\boldsymbol{\sigma}^2\left(\widehat{\boldsymbol{\beta}}\right)}{N}} = \sqrt{\frac{\left(\widehat{\boldsymbol{H}}^{-1}\right)}{N}} \\ &= \begin{bmatrix} \sigma^2(\beta_1) & \sigma(\beta_1)\sigma(\beta_2) & \cdots & \sigma(\beta_1)\sigma(\beta_n) \\ \sigma(\beta_2)\sigma(\beta_1) & \sigma^2(\beta_2) & & \sigma(\beta_2)\sigma(\beta_n) \\ \vdots & & \ddots & \vdots \\ \sigma(\beta_n)\sigma(\beta_1) & \sigma(\beta_n)\sigma(\beta_2) & \cdots & \sigma^2(\beta_n) \end{bmatrix}_{n\times n} \end{aligned} \tag{3.19}$$

where $\boldsymbol{\sigma}^2\left(\widehat{\boldsymbol{\beta}}\right)$ is the variance-covariance matrix of the parameters, $\widehat{H}^{-1}$ is the approximated inverse of the Hessian matrix, and N is the total number of observations. The diagonal elements of $\boldsymbol{\sigma}^2\left(\widehat{\boldsymbol{\beta}}\right)$ is the variances of parameters. Then, assuming the null hypothesis of β_o=0, t-statistics of each parameter can be obtained.

$$t_{\hat{\beta}_n} = \frac{\hat{\beta}_n - \beta_o}{\mathrm{SE}(\hat{\beta}_n)} \tag{3.20}$$

Eq. (3.20) denotes t-statistics of a parameter $\hat{\beta}_n$ and $n \in N$, the total number of estimated parameters. Detailed information on computing the robust t-ratio can be found in the documentation of Biogeme by Bierlaire (2016).

Please note that, while finite differences (numerical differentiation) estimate the gradient (the first-order derivative) using the difference between a certain point and the point added by a small value, the chain rule-based differentiation (AD) produces the exact derivative values. That is, the computational graph-based structures can avoid truncation and round-off errors due to numerical differentiation and accordingly improve the computational efficiency (Chapra and Canale, 2010). Table 3.2 presents the different characteristics of three estimation models.

Table 3.2. Attributes of Two Leading Estimation Packages and CG-Based Models

	CG-based Models	**Biogeme**	**Apollo**
Objective function	Log-likelihood ($\ln P_{ni}(\boldsymbol{\beta})$)		
Starting values of the parameters (MNL and NL)	$\beta_i = 0$ where $i = 0, 1, 2, \dots, n$; $\lambda_{NL} = 0.95$		
Starting values of the parameters (ICLV)	$\beta_i = 0$ where $i = 0, 1, 2, \dots, n$; $\lambda_{ICLV} = 1$; σ_i and $\zeta_i = 1$ where $i = 1, 2, 3, 4$		
Method of computing gradient derivative	Automatic differentiation through integration of domain-specific language and low-level CG layers	Chain rule of differentiation with analytical gradient	Numerical derivative using advanced extrapolation methods such as Richardson extrapolation
Optimization method	BFGS	BFGS	BFGS
Programming language	Python, C++ library	Python, C++ library	R

In general, CG and both open-source packages use the log-likelihood function as the objective function, start from the same initial values for estimation, and implement the BFGS optimizer with an approximate second-order gradient. CG and Biogeme are coded based on the Python language with underlying C++ libraries, and Apollo (0.2.4 version) is

written in the R language (computational environment: Windows Intel(R) Core (TM) i7-9750H CPU @2.60GHz, 6 Core(s), 32 GB RAM, and 500 GB SSD).

3.5. Model Estimation Results

This section provides the estimation results of MNL, NL, and ICLV models, and our focus is on the investigation of the accuracy and performance of computed gradients through various methods. The computational efficiency and numerical accuracy of the CG-based models are systematically compared to two established DCM estimation packages for MNL and NL models. Using the estimation results of ICLV, we demonstrate the ability of the proposed graph-oriented function to construct a simulation-based choice model and compare performance to the Apollo package. This research does not focus on the behavioral interpretation of the parameters (especially because NHTS data does not furnish level of service attributes critical to mode choice model specification, and the synthetic dataset is used solely for validating the CG-based models).

3.5.1. Estimation of MNL and NL with Constants Only

In Table 3.3, Part I shows the estimation results of MNL including alternative specific constants (ASCs) and their statistical properties. It is found that the graph-oriented approach shows identical estimation results when compared to Biogeme and Apollo; as noted earlier, both packages also implement the BFGS algorithm to derive the coefficients.

Part II of Table 3.3 compares numerical differences between the CG-based NL model and the benchmark packages. The calibrated coefficients (constants) from CG are consistent with the values estimated by the two packages, but the standard errors of the *Walk* constant and the *logsum* parameter $\boldsymbol{\lambda}$ show some numerical inconsistency.

In order to check the source of this inconsistency, we investigate how the packages (Biogeme and Apollo) approximate the Hessian matrix of the log-likelihood function with respect to each parameter. Biogeme aims to approximate the elements of the Hessian matrix based on chain rule differentiation (CRD) and calculate the standard errors of the coefficients.

Unlike the estimation results through CRD, the proposed modeling approach in this paper uses automatic differentiation (AD) to obtain the first order gradient of the log-likelihood function. Both approaches are based on the chain rule-based differentiation, but AD can implement intermediate variables in computing gradients, which enables the proposed model to find the analytic gradients efficiently.

Table 3.3. Model Estimation Results for MNL and NL

Part I: MNL	**DSL- based CG**			**Biogeme**			**Apollo**		
	Coef.	*Std.err*	*t-ratio*	*Coef.*	*Std.err*	*t-ratio*	*Coef.*	*Std.err*	*t-ratio*
*Driving Alone (DA; **base**)*	0	NA	NA	0	NA	NA	0	NA	NA
Shared Ride (SR)	-1.36	0.016	-84.402	-1.36	0.016	-84.402	-1.36	0.016	-84.940
Transit (TR)	-2.93	0.044	-66.547	-2.93	0.044	-66.547	-2.93	0.044	-66.510
Bike	-3.40	0.068	-50.066	-3.40	0.068	-50.066	-3.40	0.068	-50.080
Walk	-3.28	0.051	-63.870	-3.28	0.051	-63.870	-3.28	0.051	-63.780
***LL* (*initial*) // *LL* (*final*)**	-27031.930 // -16192.126			-27031.930 // -16192.126			-27031.940 // -16192.130		
AIC // BIC	32392.252 // 32426.656			32392.252 // 32426.656			32392.260 // 32426.670		
Part II: NL	**DSL-based CG**			**Biogeme**			**Apollo**		
	Coef.	*Std.err*	*t-ratio*	*Coef.*	*Std.err*	*t-ratio*	*Coef.*	*Std.err*	*t-ratio*
*Driving Alone (DA; **base**)*	0	NA	NA	0	NA	NA	0	NA	NA
Shared Ride (SR)	-1.36	0.016	-84.364	-1.36	0.016	-84.963	-1.36	0.016	-84.960
Transit (TR)	-2.92	0.044	-66.056	-2.92	0.044	-66.581	-2.92	0.044	-66.580
Bike	-3.10	0.072	-43.085	-3.10	0.073	-42.253	-3.10	0.073	-42.250
Walk	-3.12	0.059	-52.523	-3.12	0.062	-50.677	-3.12	0.062	-50.680
Logsum ($\boldsymbol{\lambda}$)	0.46	0.117	3.917	2.21*	0.622	3.556	0.45	0.127	3.560
***LL* (*initial*) // *LL* (*final*)**	-27031.94 // -16183.793			-27031.94 // -16183.78			-27031.94 // -16183.78		
AIC // BIC	32377.586 // 32420.592			32377.56 // 32420.57			32377.56 // 32420.57		

**Note*: The calculated $\boldsymbol{\lambda}$ in Biogeme is expressed as the inverse of $\boldsymbol{\lambda}$ (i.e., $1/2.21 \cong 0.45$)

Table 3.4 compares the numeric gradients extracted from two approaches (Biogeme and CG-based). In Table 3.4-Part I, we notice the gradients computed through both CRD and AD are approaching zero so that the approximated standard errors were closely identical to each other. However, as the gradient approximated by CRD in Part II (nested logit) is not sufficiently close to zero, the approximated Hessian matrix might yield different standard errors compared to the AD-based result. As shown in Eq. (3.15), the magnitude of the first-order gradients is a critical indicator for convergence, which is required to assure maximization of the log-likelihood functions (Train 2009). Please note that the approximation issue of CRD has been investigated and discussed by Brathwaite (2017) and Brathwaite and Walker, (2018a). According to Table 3.4-Part I, the absolute averages of gradients of CRD and AD are 1.32E-05 and 1.78E-09, respectively. Table 3.4-Part II shows the absolute average of the gradients of CRD is 1.16E-04 while the corresponding value for AD shows 2.83E-07. The gradients produced from both methods are significantly small, and the differences depend on the selection of stopping criteria. In other words, if we use the same stopping criteria for the estimation of gradients in both methods, the discrepancy shown in Table 4 would be vanished.

Table 3.4. Estimated Gradients Computed by Chain Rule Differentiation and Analytical Gradient (CRD+AG) and Automatic Differentiation (AD) ***through DSL***

Part I: Gradients of MNL	Chain rule differentiation and analytical gradient CRD+AG	***Automatic Differentiation (AD) through DSL***
*Driving Alone (DA; **base**)*	0	0
Shared Ride (SR)	-9.85144E-05	1.86265E-08
Transit (TR)	9.86795E-05	-1.19908E-08
Bike	-3.09112E-05	4.65661E-10
Walk	-2.21991E-05	0
Part II: Gradients of NL	CRD+AG	***AD + DSL***
*Driving Alone (DA; **base**)*	0	0
Shared Ride (SR)	1.15E-03	1.86265E-09
Transit (TR)	5.97E-04	4.08152E-07
Bike	-2.57E-05	9.76317E-07
Walk	-1.31E-03	-1.57219E-07
*Logsum (**λ**)*	1.71E-04	1.86265E-07

3.5.2. Estimation of MNL and NL with Constants and Explanatory Variables

This subsection presents estimation results for a fully specified model including explanatory variables. Specifically, five categorical variables and one continuous variable were included. The utility function of each mode is influenced by the same explanatory variables; age groups, gender, education attainment, household income and size, as well as travel time. There are 33 estimated parameters, and the detailed description of each parameter is provided in Table 3.5 and Table 3.6. Based on the log-likelihood values obtained, all methods showed similarity in terms of the estimated coefficients. On the other hand, due to the fact that the two packages used different methods to derive the gradients (numerical differentiation and chain-rule differentiation, respectively) of the parameters while the CG-based structure utilized the analytical approach (i.e., AD), we see differences

in the numeric gradients. These differences likely explain the discrepancy in standard errors and t-ratio statistics.

The gradients computed by CRD and AD are presented in Table 3.7. As expected, the gradients computed by the algorithmic differentiation are significantly closer to zero compared to the counterpart by the chain rule-based approach with different stopping criteria. In terms of the final absolute average of gradients in MNL and NL, CRD provides values of 8.72E-05 in MNL and 1.86E-04 in NL. On the other hand, the estimated gradients using AD are 9.31E-07 in MNL and 1.07E-08 in NL.

Table 3.5. Model Estimation Results for Multinomial Logit (MNL) with Explanatory Variables

Part III: MNL with explanatory variables		DSLCG -based MNL			Biogeme			Apollo		
		Coef.	*Std.err*	*t-ratio*	*Coef.*	*Std.err*	*t-ratio*	*Coef.*	*Std.err*	*t-ratio*
Drive Alone (DA; ***base***)		0.00	NA	NA	0.00	NA	NA	0.00	NA	NA
Shared Ride (SR)		-1.25	0.07	-17.69	-1.25	0.07	-17.41	-1.25	0.07	-17.38
Transit (TR)		-9.55	0.33	-29.02	-9.55	0.33	-29.12	-9.55	0.33	-29.07
Bike		-3.66	0.30	-12.15	-3.67	0.31	-11.88	-3.67	0.31	-11.86
Walk		-0.54	0.16	-3.36	-0.53	0.18	-3.05	-0.53	0.18	-3.05
SR	Gender (Male=1, Female=0)	-0.10	0.03	-3.21	-0.10	0.03	-3.18	-0.10	0.03	-3.17
	Aged 30-44 years (Yes=1, No=0)	0.12	0.04	2.85	0.12	0.04	2.80	0.12	0.04	2.79
	Aged 45-59 years (Yes=1, No=0)	-0.06	0.04	-1.48	-0.06	0.04	-1.47	-0.06	0.04	-1.47
	Education attainment: Graduate degree (Yes=1, No=0)	-0.18	0.04	-4.88	-0.18	0.04	-4.81	-0.18	0.04	-4.82
	Household income: $125,000 or more (Yes=1, No=0)	-0.06	0.03	-1.83	-0.06	0.03	-1.82	-0.06	0.03	-1.82
	Household size: Three-person or more (Yes=1, No=0)	0.11	0.03	3.38	0.11	0.03	3.32	0.11	0.03	3.32
	Natural logarithm of travel time (in minutes)	-0.02	0.05	-0.43	-0.02	0.05	-0.43	-0.02	0.05	-0.43
TR	Gender (Male=1, Female=0)	-0.15	0.04	-3.35	-0.15	0.10	-1.49	-0.15	0.10	-1.48
	Aged 30-44 years (Yes=1, No=0)	0.17	0.08	2.17	0.16	0.13	1.30	0.16	0.13	1.31
	Aged 45-59 years (Yes=1, No=0)	-0.20	0.08	-2.66	-0.21	0.13	-1.63	-0.21	0.13	-1.63

	Education attainment: Graduate degree (Yes=1, No=0)	0.45	0.08	5.54	0.45	0.10	4.32	0.45	0.10	4.32
	Household income: $125,000 or more (Yes=1, No=0)	-0.17	0.09	-1.90	-0.17	0.10	-1.67	-0.17	0.10	-1.67
	Household size: Three-person or more (Yes=1, No=0)	-0.04	0.06	-0.59	-0.04	0.11	-0.34	-0.04	0.11	-0.34
	Natural logarithm of travel time (in minutes)	4.33	0.19	22.99	4.33	0.19	22.44	4.33	0.19	22.47
Bike	Gender (Male=1, Female=0)	0.62	0.11	5.39	0.61	0.15	3.97	0.61	0.15	3.96
	Aged 30-44 years (Yes=1, No=0)	0.13	0.07	1.83	0.13	0.17	0.74	0.13	0.17	0.74
	Aged 45-59 years (Yes=1, No=0)	-0.36	0.07	-5.33	-0.36	0.18	-2.00	-0.36	0.18	-2.01
	Education attainment: Graduate degree (Yes=1, No=0)	0.55	0.08	6.87	0.55	0.14	3.93	0.55	0.14	3.93
	Household income: $125,000 or more (Yes=1, No=0)	0.06	0.08	0.80	0.06	0.14	0.41	0.06	0.14	0.41
	Household size: Three-person or more (Yes=1, No=0)	-0.16	0.04	-4.04	-0.16	0.15	-1.08	-0.16	0.15	-1.08
	Natural logarithm of travel time (in minutes)	-0.25	0.17	-1.46	-0.24	0.20	-1.16	-0.24	0.20	-1.16
Walk	Gender (Male=1, Female=0)	-0.17	0.06	-3.02	-0.17	0.11	-1.60	-0.17	0.11	-1.61
	Aged 30-44 years (Yes=1, No=0)	-0.07	0.07	-1.01	-0.07	0.14	-0.55	-0.07	0.13	-0.55
	Aged 45-59 years (Yes=1, No=0)	-0.46	0.08	-5.45	-0.45	0.13	-3.44	-0.45	0.13	-3.45
	Education attainment: Graduate degree (Yes=1, No=0)	0.27	0.07	3.88	0.26	0.11	2.32	0.26	0.11	2.32
	Household income: $125,000 or more (Yes=1, No=0)	-0.21	0.05	-4.10	-0.22	0.11	-2.00	-0.22	0.11	-2.00
	Household size: Three-person or more (Yes=1, No=0)	-0.19	0.04	-4.28	-0.20	0.12	-1.59	-0.20	0.12	-1.59

	Natural logarithm of travel time (in minutes)	-2.20	0.12	-18.78	-2.20	0.14	-16.18	-2.20	0.14	-16.12
LL (initial) // LL (final)		-27031.94 // -15553.39			-27031.94 // -15553.39			-27031.94 // -15553.39		
AIC // BIC		31170.78 // 31446.02			31170.78 // 31446.02			31170.78 // 31446.02		

Table 3.6. Model Estimation Results for Nested Logit (NL) with Explanatory Variables

Part IV: NL with explanatory variables		**DSLCG-based NL**			**Biogeme**			**Apollo**		
		Coef.	*Std.err*	*t-ratio*	*Coef.*	*Std.err*	*t-ratio*	*Coef.*	*Std.err*	*t-ratio*
Drive Alone (DA; ***base***)		0.00	NA	NA	0.00	NA	NA	0.00	NA	NA
Shared Ride (SR)		-1.25	0.07	-17.61	1.25	0.07	17.43	1.25	0.07	17.43
Transit (TR)		-9.49	0.33	-29.06	9.49	0.33	28.97	9.49	0.33	28.97
Bike		-2.70	0.37	-7.32	2.69	0.37	-7.26	2.69	0.37	-7.26
Walk		-0.70	0.18	-3.97	0.70	0.18	-3.96	0.70	0.18	-3.96
Logsum (**λ**)		0.56	0.11	5.00	1.80*	0.37	4.84	0.56	0.12	4.84
SR	Gender (Male=1, Female=0)	-0.10	0.03	-3.22	0.10	0.03	-3.18	0.10	0.03	-3.18
	Aged 30-44 years (Yes=1, No=0)	0.12	0.04	2.80	0.12	0.04	2.78	0.12	0.04	2.78
	Aged 45-59 years (Yes=1, No=0)	-0.06	0.04	-1.47	0.06	0.04	-1.46	0.06	0.04	-1.46
	Education attainment: Graduate degree Yes=1, No=0)	-0.18	0.04	-4.93	0.18	0.04	-4.85	0.18	0.04	-4.85

		Household income: $125,000 or more Yes=1, No=0)	-0.06	0.03	-1.83	-0.06	0.03	-1.82	-0.06	0.03	-1.82
		Household size: Three-person or more Yes=1, No=0)	0.11	0.03	3.37	0.11	0.03	3.33	0.11	0.03	3.33
		Natural logarithm of travel time (in minutes)	-0.02	0.05	-0.40	-0.02	0.05	-0.39	-0.02	0.05	-0.39
	TR	Gender (Male=1, Female=0)	-0.15	0.05	-3.23	-0.15	0.10	-1.52	-0.15	0.10	-1.52
		Aged 30-44 years (Yes=1, No=0)	0.16	0.08	2.11	0.16	0.13	1.30	0.16	0.13	1.30
		Aged 45-59 years (Yes=1, No=0)	-0.21	0.09	-2.45	-0.21	0.13	-1.64	-0.21	0.13	-1.64
		Education attainment: Graduate degree Yes=1, No=0)	0.44	0.05	8.34	0.44	0.10	4.30	0.44	0.10	4.30
		Household income: $125,000 or more Yes=1, No=0)	-0.18	0.08	-2.22	-0.17	0.10	-1.71	-0.17	0.10	-1.71
		Household size: Three-person or more Yes=1, No=0)	-0.04	0.04	-0.96	-0.04	0.11	-0.38	-0.04	0.11	-0.38
		Natural logarithm of travel time (in minutes)	4.30	0.18	23.46	4.30	0.19	22.38	4.30	0.19	22.38
Nested Group	**Bike**	Gender (Male=1, Female=0)	0.46	0.09	4.90	0.46	0.13	3.52	0.46	0.13	3.52
		Aged 30-44 years (Yes=1, No=0)	0.06	0.10	0.63	0.06	0.14	0.41	0.06	0.14	0.42
		Aged 45-59 years (Yes=1, No=0)	-0.40	0.12	-3.46	-0.40	0.15	-2.73	-0.40	0.15	-2.73
		Education attainment: Graduate degree Yes=1, No=0)	0.45	0.08	5.61	0.45	0.12	3.79	0.45	0.12	3.79
		Household income: $125,000 or more Yes=1, No=0)	0.00	0.08	0.04	0.00	0.12	0.03	0.00	0.12	0.03
		Household size: Three-person or more Yes=1, No=0)	-0.10	0.07	-1.53	-0.10	0.13	-0.80	-0.10	0.13	-0.80
		Natural logarithm of travel time (in minutes)	-0.68	0.22	-3.12	-0.68	0.22	-3.11	-0.68	0.22	-3.11
		Gender (Male=1, Female=0)	-0.09	0.07	-1.31	-0.09	0.10	-0.91	-0.09	0.10	-0.91

	Walk	Aged 30-44 years (Yes=1, No=0)	-0.01	0.11	-0.07	-0.01	0.12	-0.07	-0.01	0.12	-0.07
		Aged 45-59 years (Yes=1, No=0)	-0.41	0.11	-3.77	-0.41	0.12	-3.39	-0.41	0.12	-3.39
		Education attainment: Graduate degree Yes=1, No=0)	0.34	0.07	4.78	0.34	0.10	3.25	0.34	0.10	3.25
		Household income: $125,000 or more Yes=1, No=0)	-0.18	0.08	-2.27	-0.18	0.10	-1.78	-0.18	0.10	-1.78
		Household size: Three-person or more Yes=1, No=0)	-0.21	0.06	-3.28	-0.21	0.11	-1.88	-0.21	0.11	-1.88
		Natural logarithm of travel time (in minutes)	-2.02	0.13	-15.04	-2.02	0.14	-14.23	-2.02	0.14	-14.23
LL (initial) // LL (final)			-26962.012 // -15547.65			-27107.14 // -15547.65			-26962.02 // -15547.65		
AIC // BIC			31161.29 // 31445.13			31161.29 // 31445.13			31161.29 // 31445.13		

**Note*: The calculated logsum coefficient in Biogeme is expressed as the inverse of $\boldsymbol{\lambda}$ (i.e., $1/1.7968 \cong 0.56$)

Table 3.7. Gradients Estimated by Chain Rule Differentiation (CRD) and Automatic Differentiation (AD) through **DSL-Based CG**

Estimated Gradients	**Part III: Gradients of MNL**		**Part IV: Gradients of NL**	
	Chain Rule Differentiation	***AD*+DSL CG**	***Chain Rule Differentiation***	***AD*+DSL CG**
Drive Alone (DA; ***base***)	0	0	0	0
Shared Ride (SR)	-4.54E-04	-3.04E-06	-4.25E-04	3.09E-07
Transit (TR)	-3.47E-03	-2.75E-06	-2.15E-03	8.67E-08
Bike	1.43E-03	4.66E-07	-1.76E-03	1.12E-07
Walk	-8.73E-04	-6.45E-06	-4.12E-04	1.81E-07
Logsum ($\boldsymbol{\lambda}$)	NA	NA	1.30E-03	4.11E-07
Shared Ride (SR)				
Gender	5.35E-04	2.84E-07	3.96E-04	1.70E-07

Aged 30-44 years	2.35E-03	1.05E-06	8.39E-04	3.11E-08
Aged 45-59 years	2.48E-03	8.15E-07	4.98E-04	-7.12E-08
Education attainment: Graduate degree	8.36E-04	3.72E-07	9.26E-04	8.35E-08
Household income: $125,000 or more	5.04E-04	2.11E-07	6.06E-04	-6.84E-08
Household size: Three-person or more	-6.85E-04	1.18E-06	3.47E-04	-8.13E-08
Natural logarithm of travel time	-2.93E-03	2.55E-06	1.30E-03	-2.36E-07
Transit (TR)				
Gender	-9.98E-04	2.25E-06	5.82E-04	1.29E-06
Aged 30-44 years	-5.99E-04	8.00E-06	1.39E-03	5.01E-07
Aged 45-59 years	4.21E-04	6.02E-06	-5.89E-04	-1.55E-07
Education attainment: Graduate degree	-3.65E-04	1.22E-06	5.96E-04	-8.60E-07
Household income: $125,000 or more	-9.01E-04	9.93E-06	-5.88E-04	-6.74E-07
Household size: Three-person or more	-3.44E-04	9.02E-07	9.93E-04	-1.31E-06
Natural logarithm of travel time	-2.27E-03	-6.47E-06	-3.75E-04	2.17E-07
Bike				
Gender	8.69E-04	4.04E-06	-7.97E-04	5.82E-07
Aged 30-44 years	1.04E-03	9.75E-08	-3.04E-05	1.63E-06
Aged 45-59 years	-6.08E-04	6.12E-07	-2.67E-03	8.69E-07
Education attainment: Graduate degree	2.07E-04	-3.88E-06	3.00E-03	7.30E-08
Household income: $125,000 or more	-9.84E-04	5.42E-06	-1.25E-03	1.53E-07
Household size: Three-person or more	1.35E-04	5.99E-06	5.50E-04	-2.74E-06
Natural logarithm of travel time	-1.66E-03	-8.02E-06	-1.16E-04	-1.55E-07

Walk				
Gender	1.05E-03	-1.27E-06	5.64E-04	-2.08E-07
Aged 30-44 years	-1.21E-04	6.23E-06	-7.54E-04	7.29E-07
Aged 45-59 years	9.36E-04	-1.25E-05	2.96E-03	5.58E-07
Education attainment: Graduate degree	1.57E-03	4.72E-06	-1.02E-03	-2.80E-07
Household income: $125,000 or more	1.14E-03	1.30E-06	7.14E-04	5.81E-07
Household size: Three-person or more	1.05E-03	8.26E-06	2.43E-04	-1.15E-06
Natural logarithm of travel time	-2.08E-03	2.29E-06	1.26E-03	-2.24E-07

3.5.3. Computational Efficiency: MNL and NL

We now compare the computational efficiency across all methods. As seen in Figure 3.5, the CG-based models show the best computational performance, and a slight increase in running time is observed in both Figure 3.5 (a) and (b) when more parameters are added. Biogeme, which is written in Python, also provides excellent computational performance to compute a few parameters. However, for a larger number of parameters to be calibrated, the Biogeme package could yield a nonlinear increase in running time, particularly when models involve non-concave functions (two or multiple nested structures). The Apollo package coded in the R language demands significantly more computing resources. For instance, when estimating a large set of parameters (i.e., 89 parameters), the average running time of CG-based MNL and NL is 10.6 seconds. On the other hand, the average computing times for Biogeme and Apollo are 12 minutes and 35 minutes, respectively. In Figure 3.5 (b), it can be seen that the nested logit models estimated by Biogeme and Apollo packages require substantially more computational time when the set of variables becomes large.

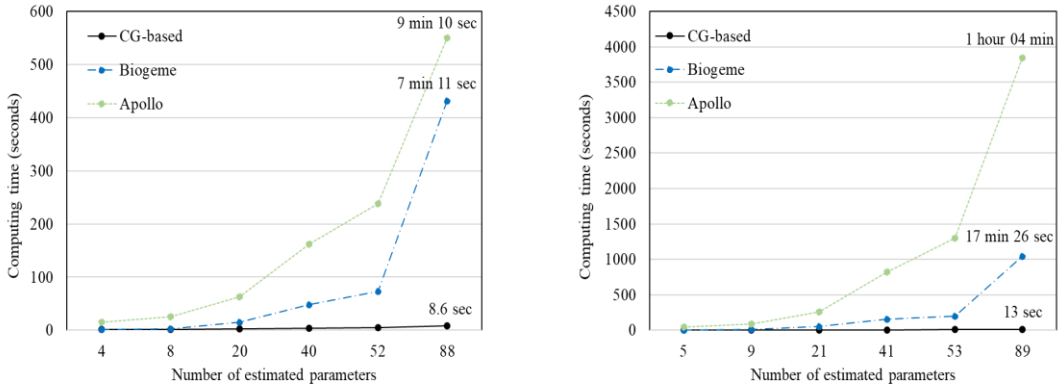


Figure 3.5. Comparison of Computation Time Between CG-Based Models, Biogeme, and Apollo

3.5.4. ICLV Model Estimation and Computational Efficiency

In this subsection, experimental results for the ICLV model are presented. The graph-oriented model and Apollo use the Monte Carlo simulation-based approach to numerically compute the ICLV function. By generating random numbers from a normal distribution, we can run the program 500 times. The specified utility function is defined by two explanatory variables and one latent variable constructed by the structural equation where it is defined by three socio-demographic characteristics. As we assume the indicators as continuous variables, components required in the normal distribution function are estimated. Table 3.8 demonstrates the ability of the CG-based approach to construct the simulation-based choice model, yielding simulated coefficients. Because the estimation involves the random sampling procedure and different methods to derive coefficients' gradients, we observe slightly different results between the CG-based ICLV, Biogeme, and Apollo. For instance, the initial log-likelihood of CG-based ICLV displays -8405.706 while Biogeme and Apollo show values of -8404.603 and -8404.237, respectively.

Table 3.8. Model estimation results for ICLV: Monte Carlo experiment

ICLV	DSLCG-based ICLV			Biogeme			Apollo		
	Coef.	*Std.err*	*t-ratio*	*Coef.*	*Std.err*	*t-ratio*	*Coef.*	*Std.err*	*t-ratio*
Parameters in the utility specification									
Drug: side-effect	-0.002	0.0002	-11.03	-0.002	0.0002	-11.1	-0.002	0.0002	-11.05
Drug: price	-0.173	0.032	-5.45	-0.173	0.032	-5.42	-0.173	0.032	-5.42
λ_{latent}	0.567	0.089	6.33	0.565	0.089	6.37	0.569	0.089	6.39
Parameters in the structural equation									
Regular user (Yes=1, No=0)	-0.677	0.072	-9.47	-0.678	0.087	-7.78	-0.677	0.087	-7.81
Education attainment: Bachelor's degree (Yes=1, No=0)	-0.253	0.054	-4.707	-0.249	0.079	-3.15	-0.248	0.079	-3.14
Aged 50 or above (Yes=1, No=0)	0.675	0.076	8.92	0.677	0.085	8.01	0.674	0.084	7.99
Parameters in measurement indicators									
ζ_{Quality}	0.562	0.044	12.7	0.557	0.045	12.3	0.564	0.046	12.4
$\zeta_{\text{Ingredients}}$	-0.565	0.043	-13.3	-0.564	0.046	-12.2	-0.564	0.046	-12.16
ζ_{Patent}	0.613	0.047	13.1	0.608	0.047	13	0.609	0.047	12.89
$\zeta_{\text{Dominance}}$	-0.400	0.036	-11.21	-0.40	0.041	-9.78	-0.401	0.041	-9.78
$\sigma_{Quality}$	1.053	0.032	33.13	1.05	0.03	34.6	1.051	0.031	34.29
$\sigma_{Ingredients}$	1.08	0.030	37.4	1.08	0.031	34.8	1.079	0.031	34.89
σ_{Patent}	1.091	0.033	32.74	1.09	0.033	33.6	1.093	0.033	33.51
$\sigma_{Dominance}$	1.047	0.025	41.57	1.05	0.027	39.5	1.047	0.027	39.48
LL (initial) // LL (final)	-8405.706 // -7552.271			-8404.603 // -7553.033			-8404.237 // -7552.271		
AIC // BIC	15132.434 // 15201.143			15134.07 // 15202.77			15132.54 // 15201.25		

In Figure 3.6, the CG-based ICLV shows the best computational performance in running Monte Carlo simulation for estimating ICLV, when compared to Biogeme and Apollo. The above limited experiments show that, when the number of simulation runs increases, the two-open source packages take more computational time than the CG-based approach using DSL.

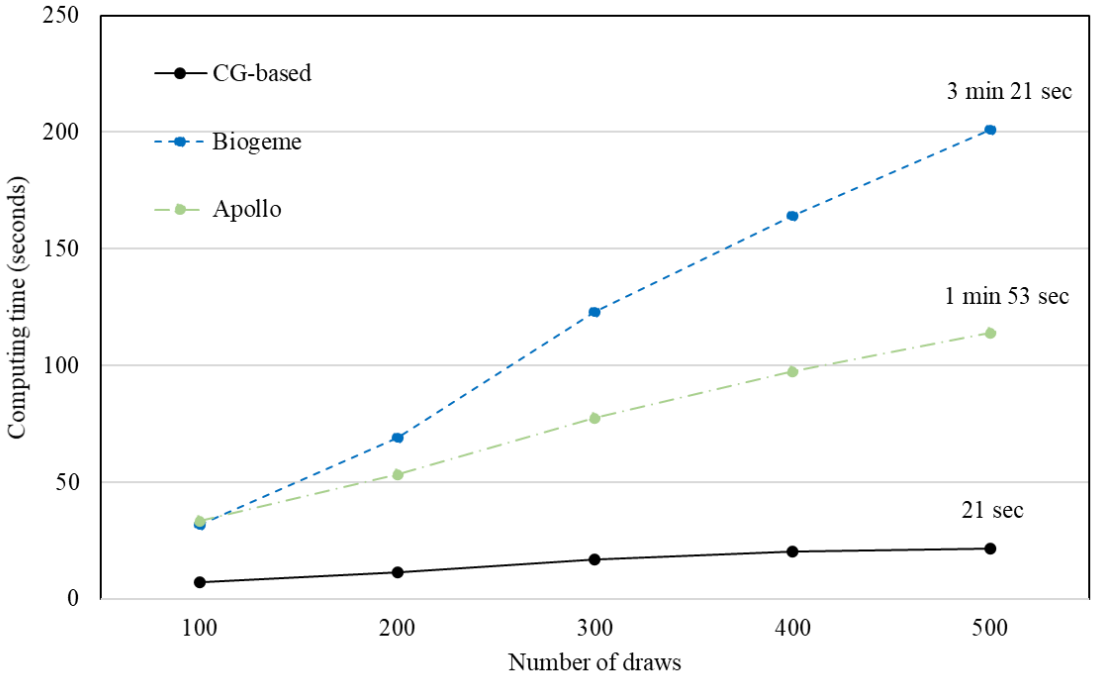


Figure 3.6. Comparison of Simulation Running Time for ICLV Estimation between DSLCG-Model, Biogeme, and Apollo

3.6. Summary and Conclusions

As the influx of real-time streaming data and new mobility technologies appears in the field of transportation, transportation planning communities are very interested in systematically integrating data-driven models and econometric models. In this paper, to bridge the gap between both methods, the functional formulation of discrete choice models is examined

in a computational graph framework, which is less known in the areas of discrete choice modeling and transportation planning, but has been widely used as underlying building blocks for deep learning packages. We hope to clearly show an implementable path to empower DCM estimation with the automatic differentiation algorithm embedded in CG, through three key findings below.

(a) A computational graph-based framework offers a highly flexible modeling method for applying the emerging techniques of deep learning in econometric methods, especially for a wide class of discrete choice models. Furthermore, CG can cover a wide range of elementary operations in its graph-oriented model representation such that researchers can easily integrate standard econometric models with machine learning algorithms that deal effectively with large amounts of time series data.

(b) In particular, for MNL and NL models, we demonstrate that CG-based learning process produces consistent estimation results compared to two leading packages, namely Biogeme and Apollo. In terms of estimating t-statistics, the *chain rule* of AD provides a robust analytical derivation, leading to converging computed gradients toward the optimality conditions. Compared to the other approximated gradient methods, the proposed approach generates high-quality estimators through a more precise Hessian matrix. Furthermore, by demonstrating the capability in the context of the ICLV modeling structure, we also show CG can be used as an effective framework in implementing extended choice models.

(c) For emerging transportation planning applications with high-dimensional survey samples and real-time big data streams, the proposed methodology holds the promise of achieving computational efficiency in handling large-scale datasets and producing rapid model updates in a cloud computing environment.

The computational graph-based architectures demonstrate the flexibility of decomposing diverse composite functions and redesigning the functions with a new functional form. In the application areas of transportation planning, researchers and planners can further use this method to improve the accuracy and time of computing/estimating systematic utility functions. As a representative example, one can better calculate the logsum term, which is widely used in practice to calculate a broad set of accessibility-oriented planning applications (Miller, 2018). One can further extend conventional modeling structures such as joint-choice models for modeling travelers' multi-dimensional choice decision-making process.

On the one hand, by building choice models through computational graph-based domain-specific languages, modelers can integrate such models easily with external deep learning architectures, leading to enhanced representation of travelers' complex activity patterns. With modeling structures capable of handling different data sources, computational graph-based modeling tools facilitate the estimation of more complex model structures, possibly improving interpretability and predictability. More precisely, the efficiency of the CG-based structures can help to rapidly estimate models that can be applied to synthetic population datasets, which are generated by microsamples and census-based marginal distributions (Ye et al., 2009; Sun et al., 2018). Additionally, since the

graph-based structure can facilitate tensor decomposition (TD) efficiently, planners are able to utilize the synthesized data and different large datasets (e.g., mobility trajectories or smart-card records), for a better understanding of travelling patterns (Sun and Axhausen, 2016).

CHAPTER 4

4. COMPUTATIONAL GRAPH-BASED MATHEMATICAL PROGRAMMING REFORMULATION FOR INTEGRATED DEMAND AND SUPPLY MODELS

As transportation systems grow in complexity, analysts need sophisticated tools to understand travelers' decision-making and effectively quantify the benefits of proposed strategies. The transportation community has developed integrated demand-supply models to capture the emerging interactive nature of transportation systems and serve diverse planning needs and encompass broader solution possibilities. Recently, utilizing the advances of machine learning techniques, researchers have also recognized the need for different computational models capable of fusing/analyzing different data sources. Inspired by this momentum, this research aims to propose a new modeling framework to analytically bridge travel demand components and network assignment models with machine learning algorithms. Specifically, to establish a consistent representation of such aspects between separate system models, we introduce several important mathematical programming reformulation techniques, namely, variable splitting and the augmented Lagrangian relaxation, to construct a computationally tractable nonlinear unconstrained optimization program. Furthermore, to find equilibrium states, we apply automatic differentiation (AD) to compute gradients of decision variables in a layered structure with the proposed model represented on the basis of computational graphs (CGs). Thus, this reformulated model offers a theoretically consistent framework to express the gap between demand and supply components and lays the computational foundation for utilizing a new generation of numerically reliable optimization solvers. Using a small example network

and Chicago skeleton transportation network, we examine the convergency/consistency measures of this new differentiable programming-based optimization structure and demonstrate the computational efficiency of the proposed integrated transportation demand and supply models.

4.1. Introduction

In this study, we aim to develop a new theoretically sound and computationally efficient framework to couple transportation demand and network models for quantifying the interaction and the impact of a wide range of congestion mitigation strategies. In particular, this research adapts a new generation of computational methods and optimization paradigm from the machine learning community, namely computational graphs (CGs) and backpropagation, to uniquely capture the layered modeling structure in the integrated demand-supply model. Furthermore, this analytically driven formulation and resulting computational architecture would extend abilities to ensure a high degree of consistency between transportation demand and supply models while sufficient descriptive capability for the interactive transportation system maintains, answering pertinent questions asked by transportation decision-makers. In the subsequent section, we first address prior work of how the community has developed integrated transportation models.

4.1.1. Integration of Transportation Demand and Network Models

As increasing complexity of planning transportation infrastructure in urban areas, transportation modeling community has dedicated great efforts to the development of integrated transportation demand and network models. Demand components are generally

modeled by trip-based or activity-based approaches, and network model structures are represented through static user equilibrium or dynamic traffic assignment. Then, to reach a consistent representation for travel time measures across different steps, a feedback loop structure is constructed from the lower-level route assignment to upper-level trip generation, destination choice and mode choice layers. To capture the full spectrum of traveler dynamics and resulting time-varying traffic congestion, recent efforts in the last 20 years have been actively devoted to integrating tour-based modeling approaches on the demand side and dynamic traffic assignment models, on the supply side (Esser and Nagel., 2001; Lam and Huang, 2003; Raney et al., 2003).

As many implementations adopt a microsimulation approach to linking transportation demand and network models, various convergence criteria are defined based on a fixed-point formulation, and the software-oriented system coupling calls for iterative solution methods using computed travel time skim profiles (Lin et al., 2008; Hao et al., 2010; Pendyala et al., 2012). For instance, Lin et al., (2008) proposed the fixed-point formulation within a variational inequality framework, with a solution based on commonly used method of successive averages (MSA). To further integrate travel demand and supply models with long-term land use evolution, Pendyala et al. (2012) developed a tightly coupled framework across three open-source packages, with various iterative processes accommodating different time updating resolutions, e.g., enroute real time decisions, post-trip learning within a day, as well as year-by-year long-term decision adjustment.

Many other related studies have also implemented a similar system integration structure with a feedback loop system between underlying subcomponents, and those

research efforts with different defining systematic structures can be examined according to three aspects, as noted by Boyce et al. (2004), mathematical formulations, solution methods, and convergence criteria (Zhou et al., 2009; Yao et al., 2014; Verbas et al., 2016; Chu, 2018). In the integration modeling studies, specifically developed for combined modal split and traffic assignment problem (CMSTA), researchers have developed different model formulations and corresponding solution algorithms to find sets of optimal path flow patterns and generalized travel costs under different criteria (e.g., user equilibrium and stochastic behavior) to quantify system behavior among system aspects and traveler choice. The recent representative studies linking travel demand and transportation supply is systematically reviewed and summarized in Table 4.1.

Table 4.1. Recent Studies for Coupling Demand-Supply Models and Corresponding Solution Algorithms

Publication	Systematic flows	Formulations for interactive systems	Gradient Computation	Step sizes for finding optimal flow patterns and travel costs	Solution algorithms for the demand and supply variables
Florian et al. (2002)	flow → cost	Variational inequality (VI)	No	Predetermined decreasing sequence	External Block Gauss-Seidel decomposition with MSA
Boyce et al. (2008)	flow → cost	Fixed point (FP)	No	Predetermined decreasing sequence	Averaging with fixed weights and MSA
Lin et al. (2008)	flow → cost	Variational inequality (VI)	No	Predetermined decreasing sequence	External sequential process with MSA
Zhou et al. (2008) and Lu et al. (2009)	flow ↔ cost	Gap function	No	Predetermined decreasing sequence	Internal circular process with MSA
Zhou et al. (2009)	flow → cost	Variational inequality (VI)	No	Rule-based algorithm	External sequential process with the self-adaptive GLP projection
Cantarella et al. (2015)	flow ↔ cost	Fixed point (FP)	No	Predetermined decreasing sequence	Internal circular process with MSA
Ryu et al. (2017)	flow → cost	Mathematical program (MP)	No	Predetermined decreasing sequence	External gradient projection (GP)
This paper	Behavior choice ↔ flow ↔ cost	Gap function-based reformulation using LR	Yes	Analytic gradients via automatic differentiation (AD)	Internal circular process with gradient-based numerical optimization (e.g., BFGS)

Table 4.1 indicates systematic directions of coupling demand and supply, formulations for interactive modeling systems, presence of gradients' information, step sizes used in updating the defined formulations, as well as the solution algorithms. Most of the existing studies on the integration of demand-supply have shown one direction or bi-directional systematic flows when finding optimal path flow patterns and generalized travel costs. In addition, different solution algorithms (e.g., MSA, gradient projection (GP), or self-adaptive gradient projection (GLP)) are proposed with predetermined step sizes or heuristic-based step sizes to find the optimal solutions in the interactive modeling systems built by VI, FP, or MP. In comparison, this paper focuses on formulating an analytically defined comprehensive framework to theoretically map between behavior choice, path flow patterns, and the travel costs and solve the optimization problem using analytic gradients obtained by automatic differentiation (AD).

4.1.2. Contributions: Analytical Gradient-Based Optimization and Computational Algorithm

While the existing literature covers a wide variety of mathematical frameworks and solutions for coupling travel demand and supply models, there are still two long-standing challenges from a mathematical modeling perspective, that is, how to efficiently evaluate and propagate derivatives of variables defined in different components, and how to recognize two essential modeling structures: (i) hierarchical layered process and (ii) complex feedback loop across different layers. Our recent work has used a CG based approach to describe various modeling elements; however, there are still two specific methodological issues to be addressed to fully utilize the new generation of optimization solvers (Wu et al., 2016; Kim et al., 2021).

- First, if a hierarchical system commonly expressed by a fixed-point approach as shown in Appendix A is computed by a feedforward approach, which calculates gradients of input variables and sends their information towards outputs, then the computational performance is significantly dependent on the number of variables to be estimated and optimized, eventually resulting in exponential computing time (Olah, 2015). In a transportation modeling perspective, a sequential structure layered by origin, origin-destination, and numerous paths and their link information includes different number of n variables in this four-layered architecture. In this case, the number of possible paths to solve the decision variables is n^4. When dealing with a larger network and set of variables n, *feedforward computation* leads to a combinatorial explosion (Olah, 2015).
- Second, a theoretical modeling framework for formulating an integrated supply-demand model is needed. Particularly, gap function-based methods to find *equivalent states* for the two components, that is transportation demand and network assignment, are required to link transportation demand and supply tightly and measure the extent of inconsistency (Lu et al., 2009).

Motivated by these two key issues, *feedforward computation* and *equivalent states*, this research focuses on establishing a novel framework for minimizing the inconsistency gap of the submodels and enhancing convergence for interactively layered structures. In particular, the concept of backpropagation, which is a core algorithm of training machine learning models (LeCun et al., 2015), is leveraged; it has been recently applied in optimizing/estimating conventional transportation planning models (Wu et al., 2018; Kim

et al., 2021). According to LeCun et al., (2015) and Olah, (2015), the backpropagation algorithm (reverse-mode differentiation) enables reduction of computing time to get gradients. For instance, the computational complexity of finding optimality in the four-layered structure with n variables can be $4n$, decreasing the cost of calculation as well as solving the *feedforward computation* issue.

To execute this algorithm, the integrated modeling framework is reformulated by a graph-oriented programming language available to derive gradients of the decision variables using automatic differentiation (AD), where the gradients of given functions are computed based on the chain rule. In order to systematically describe the inherent connections between the chain rule and AD, we provide detailed explanations in Appendix B. The details of the advantages of this algorithm can be found in Wright and Nocedal, (1999) and Baydin et al., (2017). Furthermore, to achieve the top-down process and discover *equivalent states*, a new constraint for tightly bridging demand/supply components is identified using the variable splitting method (Ortuzar and Willumsen, 2001). To find optimal values for network equilibrium states and the linking constraints, the objective function is formulated by the augmented Lagrangian relaxation with the computational graphs.

This research highlights the benefits of automated calculation of gradients in a complex and nonlinear structure with multiple layers of composite functions, and the iterative computing of step size is handled through the underlying solvers, namely TensorFlow (Abadi et al., 2016). Thus, we shift more focus on model reformulation to comprehensively incorporate network flow, utility, choice probability at different choice

dimensions, while still taking advantage of emerging computing architecture. To demonstrate the unconstrained non-linear optimization program, a simple case study and a real network (Chicago transportation network) are examined.

The remainder of this paper is organized as follows. Section 4.2 presents the problem statement. Section 4.3 describes the formulation of variational inequality for DUE and fixed-point equations to sequentially link each interactive variable used in the model. The subsequent section shows the solution algorithm (variable splitting and Lagrangian relaxation applied in reverse-mode differentiation (i.e., AD)). In Section 4.5, the performance of the integrated transportation model is evaluated using examples of traffic networks. Lastly, Section 4.6 summarizes our contributions and introduces the extension of the current version.

4.2. Problem Statement and Conceptual Reformulation Framework

4.2.1. The Integrated Demand-Supply Framework

The deep integration of travel demand and network models is of much interest, and interested readers are referred to a recent effort along this line as part of the SHRP II C10 project (Smith et al., 2018). Typically, the demand side component shares a list of individual trips to the supply side model, and the aggregated level of service (LOS) in the form of travel time matrix (i.e., skim) for all possible trips across OD pairs are fed back from the supply side model to the demand side. Through multiple iterations of flow adjustment, the ultimate goal is to reduce inconsistent time-of-day outcomes between the experienced travel time from the demand side and the expected travel time from the supply side. Theoretically, the linkage between different components should be clearly defined,

within the temporal equilibrium for the demand side and behavioral consideration on the supply side. Without loss of generality, this research aims to find stable trip demand pattern and equilibrated path flows that minimize system-wide inconsistency measures.

Mathematically, given a transportation network with a set of nodes N, and a set of links A, the problem aims to find a consistent traveler behavior and network performance solution subject to several demand and supply constraints. In the demand-side network, nodes are designated as activity locations of traffic analysis zones for the high-level traveler behavioral model, and modeling links indicate the direction of the choice behavior. In the supply-side network, the system performance associated with link traveling costs and path flows is computed on a given physical road or multimodal network. The accessibility/travel costs between OD pairs are selected as the key consistency/convergence criterion in the demand-supply integration under consideration

The conceptual model structure is shown in Figure 4.1, which denotes the layered system with a set of physical nodes n $(n \in N)$, a set of physical links $l \in A$, as well as a set of links related to the mode choice behaviors. This sequential modeling data flow starts from origin node $\boldsymbol{o}$. Based on the scale of accessibility costs ($c_{od(\bar{m})}$ and $c_{od(m\neq\bar{m})}$), $f_{od(\bar{m})}$ denotes the number of trips, selected by trip makers with alternative modes $\bar{m}$. The distribution of path/link volumes can also be quantified in $x_{odp_f(\bar{m})}$ and $x_{odp_a(\bar{m})}$, path flows on freeway corridor f and arterial corridor a; inside the path flows, the link flows are mapped by an incidence matrix, and the Bureau of Public Roads (BPR) function are applied to compute their link travel times. t_{odp_f} and t_{odp_a} are travel time variables corresponding to each path alternative. Travel cost $c_{od(\bar{m})}$ on a certain OD pair should be

computed to meet the DUE condition. Lastly, to minimize the gap of the shared variables $c_{od(\bar{m})}$ between both demand and supply side, the feedback loop system is activated, leading to a tightly coupled model system. This simplified connectivity aims to demonstrate a parsimonious structure to bridge behavioral elements and network performance elements.

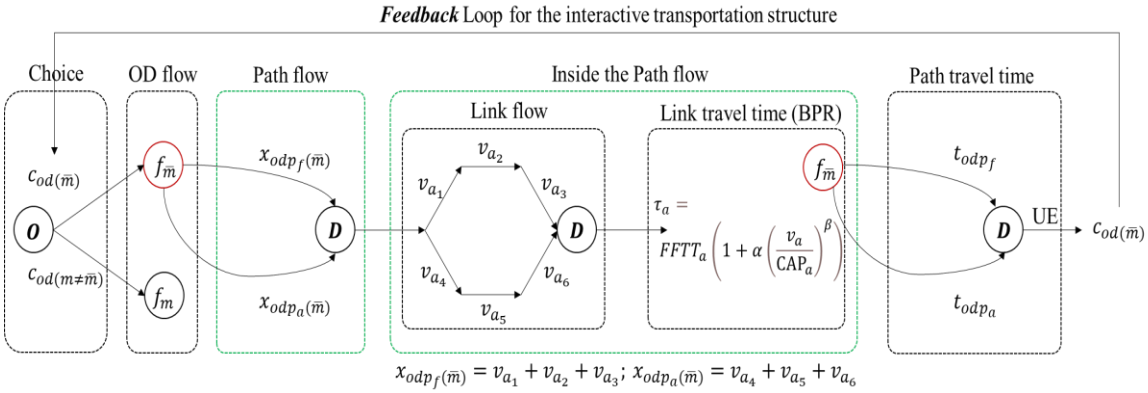


Figure 4.1. Conceptual Illustration of Integrated Supply-Demand System in a Simple Network

In the above small example, there are two essential decision variables: accessibility/travel costs $c_{od(\bar{m})}$ and path flow $x_{odp_f(\bar{m})}$ and $x_{odp_a(\bar{m})}$, dependent on the number of available paths and the number of OD pairs. It should be noted that, there exists a set of composite functions to compute travel cost $c_{od(\bar{m})}$, jointly determined by the choice model and the network models. Specifically, t_{odp_f} and t_{odp_a} are determined by the volume delay or link performance function and a path-link incidence matrix. The generic formulation of the simplified demand-supply integration program is provided in the following subsection. The indices, variables, and parameters used in the formulation are listed in Table 4.2. Before discussing the equations for the demand side and the supply side,

we would like to first introduce a variable splitting method to reformulate the feedback loop system.

Table 4.2. Notation

Network information $G = (N, A)$ and indices:	
T_o	Total demand on origin node o
TD_{od}	Total demand on destination node d
o	Subscript for an origin node, $\boldsymbol{o \in N}$
d	Subscript for a destination node, $\boldsymbol{d \in N}$
p	Subscript for a path, $p \in P(o,d)$
$\bar{p}$	Subscript for a path to represent the last path variable of a set of path variables $p \neq \bar{p}$
(m)	Alternative modes m (a set of alternative modes); $m = \bar{m}$ (auto mode) and $m \neq \bar{m}$ (transit)
CAP_a	Link capacity on link a
a	Index for a link, $a = 1,2, \dots, \lvert A \rvert$
(l)	Superscript for a layer index
Input parameters/values:	
θ	Dispersion parameter
λ, ρ	Lagrange multiplier and penalty term
δ_{odpa}	Incidence matrix for the linkage between path layers and link layers
α, β	The BPR function parameters (e.g., $\alpha = 0.15$ and $\beta = 4$)
$FFTT_a$	Free flow travel time on a link, a
Decision and intermediate variables:	
$f_{od(\bar{m})}$	Number of trips from node o to node d by mode $\bar{m}$
$P(o,d)$	Set of all feasible paths for a given pair (o,d)
$x_{odp(m)}$	Number of trips from node o to node d in path p by mode m
$x_{odp(\bar{m})}$	Number of trips from node o to node d in path p by mode $\bar{m}$
t_{odp}	Path travel time cost from node o to node d in path p
$c_{od(m)}$	Common variable for supply side and demand side in a given od pair with mode m
$c^s_{od(m)}$	Least travel cost, min t in a given od pair (supply side) with mode m
$c^d_{od(m)}$	Accessibility cost in selecting alternative modes (demand side) with mode m
v_a	Link flows on link a
τ_a	Link travel time on link a

4.2.2. Variable Splitting Method for Enabling Problem Decomposition of Feedback Loops

In general, successive feedback loops between different elements are used to model mutual-causal interaction in transportation networks and ensure equilibrium states (Lin et al., 2008; Pendyala et al., 2012; Chu, 2018). Specifically, the potential travel demand is estimated by econometric models, and it is sent to network models to measure network performance such as level of service (LOS). Then, the information returns to the demand side to re-generate the potential trips and their attributes. This iterative procedure is continuously performed until integrated models meet convergence criterion. From the perspective of problem decomposition, we highlight the need to mathematically **"decouple"** common variables (i.e., costs) used in both sub-problem models (demand and supply).

A key principle in many large-scale system optimization applications is to first decompose the constraints into "easy" vs. "difficult" constraints and then dualize the "difficult" constrains to enable the resulting components or subproblems can be efficiently solved (Fisher et al., 1997). On the other hand, variable splitting is another important problem decomposition method, which splits the common variables and create an additional coupling constraint between "duplicated" variables, and then dualize the newly introduced coupling constraint in the objective function to generate two "easy-to-solve" subproblems (Guignard, 2003). Mathematically, consider a general mathematical function to link the two demand and supply subsystems with essential constraints. The integration optimization problem can be described as $\min_x\{I(x)|Ax \leq b, Cx \leq d, x \in X\}$ where $I(x)$ indicates a function for the subsystem integration, x is a vector of demand and supply variables that we need to find optimal solutions, and the constraints, $Ax \leq b$ and $Cx \leq d$,

are associated with the demand and supply sides respectively. Based on the concept of the variable splitting method, we can decompose part of common cost variable x used in both subsystems/constraints as follows: $x \rightarrow x_d$ and x_s, which leads to $\min_{x_d, x_s}\{I(x)|Ax_d \leq b, Cx_s \leq d, x_d \in X \text{ and } x_s \in X, x_d = x_s\}$.

4.3. Formulation of the Integrated Demand-Supply Optimization (IDSO) Program

This section describes a sequence of reformulation steps for IDSO, with the augmented Lagrangian relaxation function as the core solution algorithm in a stepwise manner.

M0: Original Form with Objective Functions and Constraints

Without loss of generality, this paper uses a logistic regression model to compute the number of destination choices and the number of trips associated with alternative modes. For instance, the destination choices and the number of trips selecting auto mode are expressed as follows:

$$TD_{od} = T_o \times \frac{e^{U_{(j)}}}{\sum_k e^{U_{(k)}}}, \forall o \tag{4.1a}$$

$$f_{od(\bar{m})} = TD_{od} \times \frac{e^{-\theta c_{od(\bar{m})} + U_{(\bar{m})}}}{e^{-\theta c_{od(\bar{m})} + U_{(\bar{m})}} + \sum_{m \neq \bar{m}} e^{-\theta c_{od(m)} + U_{(m)}}}, \forall o, d \tag{4.1b}$$

Eq. (4.1a) represents the product of the total demand on origin o and the choice probability of a destination and derives the number of the total demand on destination d, TD_{od}. $U_{(j)}$ is the utility function, which can be defined by spatial separation between origin o and destination d and attractions at d (Fotheringham 1986).

Depending on the accessibility cost $c_{od(\bar{m})}$ and dispersion parameter θ, the potential number of trips/riders in selecting auto modes can be computed. With the assumption that other behavioral parameters (i.e., socio-demographic characteristics) are

given by choice modeling estimators (e.g., Bierlaire 2003; Kim et al., 2021), systematic utility function $U_{(\bar{m})}$, which measures the satisfaction in selecting the alternative mode, is assumed to be constant values for simplicity. Eq. (4.1b) denotes an essential form of the disutility/utility function, leading to the number of trips, $f_{od(\bar{m})}$, from origin o to destination d with mode $\bar{m}$. It should be noted that in order to simplify the complexity of decision-making processes, we first consider the integration of Eq. (4.1b) with a network model.

Now, in order to assign the trip information in a network, the variational inequality (VI) condition is formulated to find DUE states (Zhou et al., 2009; Lu et al., 2009). With path flow $x_{odp(\bar{m})}$ and path travel cost $t_{odp(\bar{m})}$, optimal solutions that satisfy the Wardrop's principle can be obtained:

DUE condition:

$$x_{odp(\bar{m})}\left(t_{odp(\bar{m})} - c_{od(\bar{m})}\right) = 0, \forall o, d, p \in P(o, d) \tag{4.2}$$

Minimum cost definitional constraint:

$$t_{odp(\bar{m})} - c_{od(\bar{m})} \geq 0, \forall o, d, p \in P(o, d) \tag{4.3}$$

where $c_{od(\bar{m})}$ is the least travel time cost traveled from o to d with mode $\bar{m}$. The practical interpretation of Eq. (4.2) and (4.3) is that traffic path flows are assigned into least-cost paths until reaching equilibrium states across all possible paths (Lo and Chen, 2000). Moreover, to assure the path flow conservation and the non-negativity for path flows, the additional constraints are written as follows:

Flow conservation and non-negativity constraints:

$$\sum_{p \in P(o,d)} x_{odp(\bar{m})} = f_{od(\bar{m})}, \forall o, d \tag{4.4}$$

$$x_{odp(\bar{m})} \geq 0, \forall o, d, p \in P(o, d) \tag{4.5}$$

Eq. (4.4) indicates the summation of the path flows is equal to the number of trips obtained from Eq. (4.1b). In order to clearly express the complex composite functions involved with the constraints, the fixed point-based approach is employed. Next, the path travel time costs, $t_{odp(\bar{m})}$, are calculated, and an incidence matrix, which connects path layers and link layers, is embedded to derive link performances and path travel time costs.

$$v_a = \sum_o \sum_d \sum_{p \in P(o,d)} x_{odp(\bar{m})} \times \delta_{odpa} \tag{4.6a}$$

$$\tau_a = FFTT_a \left(1 + \alpha \left(\frac{v_a}{\mathrm{CAP}_a}\right)^{\beta}\right), \forall a \tag{4.6b}$$

$$t_{odp(\bar{m})} = \sum_a \delta_{odpa} \times \tau_a \,, \forall o, d, p \in P(o,d) \tag{4.6c}$$

$$c_{od(\bar{m})} = \min_{p \in P(o,d)} \left\{ t_{odp(\bar{m})} \right\} \tag{4.6d}$$

Eqs. (4.6a) – (4.6d) illustrate the stepwise computation to obtain the final layer of supply-side variable as the path travel time costs. The first step computes the link volumes using the results of Eq. (4.1b) and the path-link incidence matrix, δ_{odpa} (if link a is on path p, then 1 otherwise 0). The subsequent step uses the Bureau of Public Roads (BPR) function to obtain the link travel time τ_a, from origin node o to destination node d along path p. Lastly, by multiplying the link-path incidence matrix δ_{odpa} and τ_a, we can derive the path travel time costs. This layered composite function then evaluates the least travel time cost $c_{od(\bar{m})}$ using $\min_{p \in P(o,d)} \left\{ t_{odp(\bar{m})} \right\}$, which can be explained as a fixed-point formulation (Appendix A). Furthermore, in order to convert constrained program with equality constraints to a unconstrainted program, a reduced gradient method for a typical path flow-based formulation (Jayakrishnan et al., 1994; Chen et al., 2001) is used in this study. Instead of expressing the flow conservation in Eq. (4.4), we consider non-shortest

path set as non-basic variable, the shortest path flow as the basic variable, so that the number of variables can be reduced through the linear composition function $x_{od\bar{p}(\bar{m})} = f_{od(\bar{m})} - \sum_{p\in P(o,d);p\neq\bar{p}} x_{odp(\bar{m})}$.

Thus, the constraints defined in Eqs. (4.1b) – (4.6d) in the original nonlinear optimization program is now expressed as a set of composite functions which are all embedded in the objective function.

min	Eq. (4.2) written as $\sum_o \sum_d \sum_{p\in P(o,d)} x_{odp(\bar{m})}\left(t_{odp(\bar{m})} - c_{od(\bar{m})}\right)$	(4.7a)
variable-coupling equations	Eq. (4.1b) $= f_{od(\bar{m})}$	(4.7b)
	Eq. (4.4) written as $x_{od\bar{p}(\bar{m})} = f_{od(\bar{m})} - \sum_{\substack{p\in P(o,d)\\ p\neq\bar{p}}} x_{odp(\bar{m})}, \forall o,d$	(4.7c)
	Eq. (4.3)	
	Eq. (4.5)	
	Eq. (4.6a) – (4.6d)	

In this optimization model, Eq. (4.7a) is the primal objective function with a number of variable-coupling functions to link the demand model and the network model. Eqs. (4.6a) – (4.6d) show link volume v_a is defined by the summation of the corresponding path flows, $x_{odp(\bar{m})}$ with path-link incidence matrix δ_{odpa}. Then, link travel time τ_a is computed by the BPR function, and the path flow travel time $t_{odp(\bar{m})}$ is obtained by the product of τ_a and the link-path incidence matrix. Please note that to fulfill the shortest path-based assignment, we use the minimum function to find the lower cost in $t_{odp(\bar{m})}$, finding the least travel cost $c_{od(\bar{m})}$ (Jayakrishnan et al., 1994).

It is important to remark that the proposed nonlinear model **M0** can be extended to a rich set of demand and supply forms, in different ways of coding complex nonlinear functional forms. *First*, the relatively simple destination choice model (i.e., travel demand

component) can be extended into more realistic behavior models. For instance, the choice modeling structure can describe more details of travelers' decision-making patterns on the basis of the activity-based framework studied by Bowman (1998). Using the concept of Downward (conditionality) and Upward (accessibility), the activity-based modeling structure can be designed by the top-down modeling formulation. Specifically, by synthesizing different utility functions defined in the hierarchically layered system (long-term decision and daily scheduling decision including tours and trip/stop), modelers can handle the conditionality of each choice model. Then, defining log-sums variables (i.e., the expected maximum utility function), we can also design the accessibility between each layer. Such a nested system of discrete choice models can model travelers' activity sequences theoretically where we can use the analytical behavioral models.

Second, the volume-delay function (VDF), expressed as the Bureau of Public Roads (BPR) function, can be formulated as queue-evolution based VDF (Belezamo, 2020) and other types of deterministic fluid based models using polynomial time-dependent arrival rates (Newell, 1968; Newell, 1982). According to the functional forms proposed, the closed form solutions from the deterministic fluid-based queuing models can be systematically defined, through computing the average system-wide delay as a function of queued demand and ultimate capacity, which can help us incorporate more realistic system performance models.

In a nutshell, the detailed examination of such nonlinear functional forms found in the travel demand modeling and the network modeling can be applicable in our approach and will be conducted.

M1: Variable Splitting

In the **M0** module, the common variable appears in both systems, which means $c_{od(\bar{m})}$ is indicated as accessibility cost in the choice model and travel cost in the network model. To handle the common variable simultaneously, the variable splitting method is used: $c_{od(\bar{m})} \rightarrow c^{d}_{od(\bar{m})}$ and $c^{s}_{od(\bar{m})}$.

min	Eq. (4.7a) written as $\sum_o \sum_d \sum_{p \in P(o,d)} x_{odp(\bar{m})}\left(t_{odp(\bar{m})} - c^{s}_{od(\bar{m})}\right)$	(4.8a)
variable-coupling equations	Eq. (4.3) written as $t_{odp(\bar{m})} - c^{s}_{od(\bar{m})} \geq 0, \forall o, d, p \in P(o,d)$	(4.8b)
	Eq. (4.6d) written as $c^{s}_{od(\bar{m})} = \min\left\{t_{odp(\bar{m})}\right\}$	(4.8c)
	$c^{d}_{od(\bar{m})} - c^{s}_{od(\bar{m})} = 0$: variable splitting constraint	(4.8d)
	Eq. (4.7b)	
	Eq. (4.7c)	
	Eq. (4.5)	
	Eq. (4.6a) – (4.6c)	

M2: Dualize the Constraints for the Unconstrained Optimization Program

To find optimal solutions of the constrained optimization subject to equality and inequality constraints, we reformulate **M1** as an unconstrained optimization problem using the augmented Lagrange relaxation method. The three components are dualized: the VI condition, the fixed-point formulations (demand-supply interaction functions), and the variable splitting constraints. Eq. (4.8d) and Eq. (4.5) are dualized using Lagrangian multipliers and quadratic penalty functions. Furthermore, in order to compute composite variables such as $c^{s}_{od(\bar{m})}$, we use the constraints in Eqs. (4.6a) – (4.6c), and Eq. (4.8c).

$$\mathcal{L}_{AL}\left(x_{odp(\bar{m})}, c^{d}_{od(\bar{m})}; \boldsymbol{\lambda}, \boldsymbol{\rho}\right) \tag{4.9}$$

$$= \sum_{o}\sum_{d}\sum_{p\in P(o,d)} x_{odp(\bar{m})}\left(t_{odp(\bar{m})} - c^{s}_{od(\bar{m})}\right)$$
$$+ \sum_{o}\sum_{d}\left[\lambda_v\left(c^{d}_{od(\bar{m})} - c^{s}_{od(\bar{m})}\right) + \frac{\rho_v}{2}\left(c^{d}_{od(\bar{m})} - c^{s}_{od(\bar{m})}\right)^2\right]$$
$$+ \sum_{o}\sum_{d}\sum_{p\in P(o,d)}\left[\lambda_x\left(x_{odp(\bar{m})}\right) + \frac{\rho_x}{2}\left(x_{odp(\bar{m})}\right)^2\right]$$

M3: Computational graph-based framework for the sequential architecture

Eq. (4.9) presents the variable splitting-based augmented Lagrange relaxation framework with multipliers. In order to find optimal solutions efficiently in this layered structure with a large set of decision variables, we employ the principle of dynamic programming (DP) through the backpropagation mechanism. Eq. (4.9) is now converted as a computational graph-based framework, translating the optimization problem into a sequentially layered architecture. The unconstrained program can be written as a system of nested composite functions (Recht, 2016):

$$\begin{aligned}\mathcal{L}_{CGAL}&\left(x_{odp(\bar{m})}, c^{d}_{od(\bar{m})}; \boldsymbol{\lambda}, \boldsymbol{\rho}\right)\\ &= \sum_{o}\sum_{d}\sum_{p\in P(o,d)} x_{odp(\bar{m})}\left(t_{odp(\bar{m})} - c^{s}_{od(\bar{m})}\right)\\ &+ \sum_{o}\sum_{d}\sum_{p\in P(o,d)}\left[\lambda_v\left(c^{d}_{od(\bar{m})} - c^{s}_{od(\bar{m})}\right)\right.\\ &\left.+ \frac{\rho_v}{2}\left(c^{d}_{od(\bar{m})} - c^{s}_{od(\bar{m})}\right)^2\right]\\ &+ \sum_{o}\sum_{d}\sum_{p\in P(o,d)}\left[\lambda_x\left(x_{odp(\bar{m})}\right) + \frac{\rho_x}{2}\left(x_{od\bar{m}}\right)^2\right]\end{aligned} \tag{4.10}$$

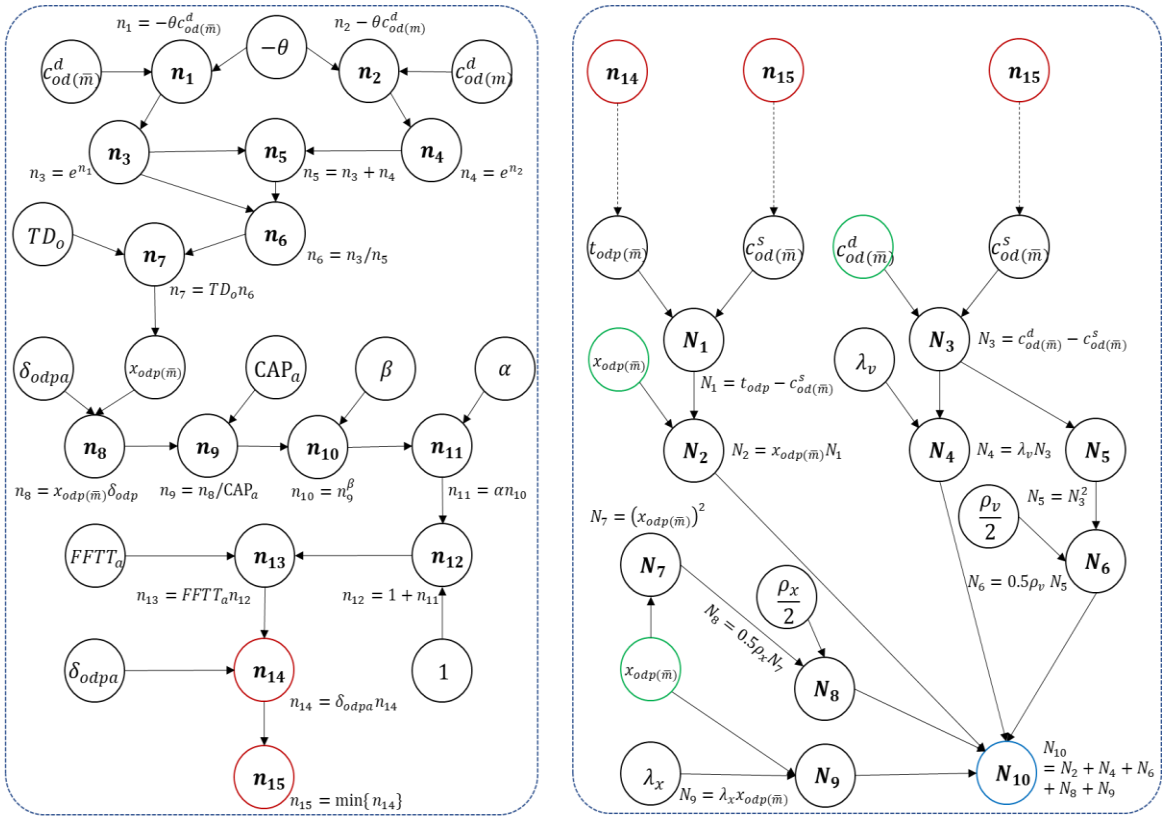
subject to

$$f_{od(\bar{m})} = g_1\left(c^{d}_{od(\bar{m})}, \theta, TD_o\right) \tag{4.11a}$$
$$x_{odp'(\bar{m})} = g_2\left(f_{od(\bar{m})}\right), p' \in \{p, p' \neq \bar{p}\} \tag{4.11b}$$
$$x_{od\bar{p}(\bar{m})} = g_3\left(f_{od(\bar{m})}, x_{odp'(\bar{m})}\right), \bar{p} \in \{p, \bar{p} \neq p'\} \tag{4.11c}$$
$$v_a = g_4\left(x_{odp'(\bar{m})}, x_{od\bar{p}(\bar{m})}, \delta_{odpa}\right) \tag{4.11d}$$
$$\tau_a = g_5\left(v_a, FFTT_a, \alpha, \beta, \mathrm{CAP}_a\right) \tag{4.11e}$$
$$t_{odp(\bar{m})} = g_6\left(\tau_a, \delta_{odpa}\right) \tag{4.11f}$$
$$c^{s}_{od(\bar{m})} = g_7\left(t_{odp(\bar{m})}\right) \tag{4.11g}$$

where Eqs. (4.11a) – (4.11g) indicate the variable-linking functions to compute the dualized optimization problem, $\mathcal{L}_{CGAL}$, and these functions nicely enable the powerful capability of constructing feedforward and backward computations through automatic differentiation (AD). Thanks to this computational graph architecture, the gradients of Eq. (4.10), which help discover optimal solutions, can be calculated in a manner of dynamic programming (DP) and symbolical gradients. In other words, this critically important reformulation technique uses the backward propagation principle in widely available machine learning packages, further reducing the efforts of computing gradients. The details of computational graph-based functions expressed in Eqs. (4.11a) – (4.11g) are illustrated below.

$g_1(\cdot)$	= Eq. (4.1b)
$g_2(\cdot)$	= Fixed and non-shortest path flows, $x_{odp'\bar{m}} \geq 0$
$g_3(\cdot)$	= Fixed and shortest path flows, $x_{od\bar{p}\bar{m}} \geq 0$
$g_4(\cdot)$	= Eq. (4.6a)
$g_5(\cdot)$	= Eq. (4.6b)
$g_6(\cdot)$	= Eq. (4.6c)
$g_7(\cdot)$	= Eq. (4.8c)

The recent approaches of applying the machine learning algorithm to compute higher-order gradients efficiently can be found in studies (Wu et al., 2016; Kim et al., 2021) for different problems such as OD demand estimation and discrete choice model calibration.



(a) CG-based composite function for $t_{odp(\bar{m})}$ and $c^s_{od(\bar{m})}$ (b) CG-based augmented Lagrangian structure

Figure 4.2. Illustration of the Sequential Structure for the Composite Values and the Unconstrained Optimization Problem

As shown in Figure 4.2, the sequential structure to define the composite terminologies and the unconstrained optimization problem which presents the augmented Lagrangian expression in Eq. (4.10) is described. For the simplification of expressing the graph-oriented structure, we consider one OD pair with two paths. The number of the decision variables is one cost variable $c^d_{od(\bar{m})}$ and one path flow variable $x_{odp(\bar{m})}$ when applying the reduced gradient algorithm, Eq. (4.7c). In this figure, the $\boldsymbol{n_7}$ node defines the formulation defined in Eq. (4.1b), deriving the number of trips on the OD pair. $\boldsymbol{n_{14}}$ and $\boldsymbol{n_{15}}$ nodes represent path travel time cost $t_{odp(\bar{m})}$ and least travel time cost $c^s_{od(\bar{m})}$ and are substituted into Figure 4.2 – (b) as input variables.

Remark 1. In order to approximate the perfect traffic assignment solution through modeling the path selection, we use the minimum functional form in path travel time cost vectors $t_{odp(\bar{m})}$, selecting the least travel time cost value $c^s_{od(\bar{m})} = \min\{t_{odp(\bar{m})}\}$. Then, this computational graph-based augmented Lagrangian relaxation (CGLR) optimization function is expressed as:

$$\begin{aligned} \mathcal{L}_{CGAL} &= N_2 + N_4 + N_6 + N_8 + N_9 \\ &= x_{odp}N_1 + \lambda_v N_3 + \frac{\rho_v}{2}N_5 + \lambda_x x_{odp(\bar{m})} + \frac{\rho_x}{2}N_7 \\ &= \frac{x_{odp(\bar{m})}\left(t_{odp(\bar{m})} - c^s_{od(\bar{m})}\right) + \lambda_v\left(c^d_{od(\bar{m})} - c^s_{od(\bar{m})}\right) +}{\frac{\rho_v}{2}\left(c^d_{od(\bar{m})} - c^s_{od(\bar{m})}\right)^2 + \lambda_x x_{odp(\bar{m})} + \frac{\rho_x}{2}\left(x_{odp(\bar{m})}\right)^2}; \end{aligned} \tag{4.12}$$

4.4. Solution Algorithm

After translating the proposed optimization problem into the directed computational graph, the partial derivatives of Eq. (4.12) are calculated. Please note that the simplified case is still used in order to intuitively explain the chain rules using gradients. With the following assumption mentioned below, we implement AD on the basis of the chain rule and intermediate nodes defined in the graph structure. As stated in Bartholomew-Biggs et al., (2000), AD operates the chain rule to generate analytic derivatives with respect to the given function, which guarantees the accuracy of computed gradients. In addition, by building intermediate nodes in the original function (e.g., nodes in Figure 4.2 such as N_2, N_4 or N_6 linking between input nodes and the output nodes), it can decompose functions and improve computing performance for the gradient-oriented optimization.

Assumption 1. The function defined in Eq. (4.10) is a continuous function at any points.

Assumption 2. The defined optimization problem has the local minimum.

With Eq. (4.10) and its computation graph in Figure 4.2. The partial derivatives of augmented LR with respect to $x_{odp(\bar{m})}$ can be expressed as follows:

$$\begin{aligned}\frac{\partial \mathcal{L}_{CGAL}}{\partial x_{odp(\bar{m})}} &= \frac{\partial N_{10}}{\partial N_9}\frac{\partial N_9}{\partial x_{odp(\bar{m})}} + \frac{\partial N_{10}}{\partial N_8}\frac{\partial N_8}{\partial N_7}\frac{\partial N_7}{\partial x_{odp(\bar{m})}} + \frac{\partial N_{10}}{\partial N_2}\frac{\partial N_2}{\partial x_{odp(\bar{m})}} \\ &= \lambda_x + \rho_x x_{odp(\bar{m})} + N_1 \\ &= \lambda_x + \rho_x x_{odp(\bar{m})} + t_{odp(\bar{m})} - c^s_{od(\bar{m})}\end{aligned} \tag{4.13}$$

Similarly, based on the chain rule method and the graph in Figure 4.2, we can compute the gradients with respect to $c^d_{od(\bar{m})}$, and this computation is achieved by the backpropagation algorithm (reverse-mode automatic differentiation). Because of this feedforward expression and the backpropagation computation, the efficiency of computing a larger set of decision variables defined in the proposed model can be guaranteed (Olah, 2015).

For the estimation of different econometric choice models defined as non-concave logistic formulations (e.g., nested logit model), a quasi-Newton method (BFGS) is employed. Now, once finding optimal solutions via the numerical optimizer, the primal problem and its constraints are computed to confirm consistency/convergence criterion. If not able to reach the satisfactory conditions, Lagrangian multipliers are updated. Using the exemplified equation, the updating process can be processed by:

$$\lambda_x^{k+1} = \max\{0, \lambda_x^k + \rho_x(x^k_{odp\bar{m}})\} \tag{4.14}$$

$$\lambda_v^{k+1} = \max\{0, \lambda_v^k + \rho_v(c^d_{od(\bar{m})} - c^s_{od(\bar{m})})\} \tag{4.15}$$

Eqs. (4.14) – (4.15) denote the Lagrangian penalties involved with path flow positivity and demand-supply consistency, which can be renewed by the defined constraints and the quadratic penalty parameters, ρ_x and ρ_v. Updating these variables, the

building block language-based program resolves the optimization problem by finding the minimum gap for UE condition, demand-supply, and positivity for path flows. This iterative process is completed when reaching the convergence criterion; the solving process for CGAL is illustrated below.

Algorithm: Augmented Lagrangian method-based integration of demand and supply

Step 1: Initialization
Load demand and network data
Initialize iteration number $k = 0$;
Initialize variables $c_{od(\bar{m})}^{s(k)}$, $t_{od(\bar{m})}^{(k)}$, $x_{odp(\bar{m})}^{(k)}$, and $c_{od(\bar{m})}^{d(k)}$
Initialize Lagrangian multipliers $\boldsymbol{\lambda}^k$ and set up quadratic penalty parameters $\boldsymbol{\rho}$
Step 2: Minimize the augmented Lagrangian method over each component
Step 2.1: Call the function $\mathcal{L}_{CGAL}(x_{odp(\bar{m})}, c_{od(\bar{m})}^{d}; \boldsymbol{\lambda}, \boldsymbol{\rho})$ defined in Eq. (4.10)
Step 2.2: Compute gradients of Eq. (4.10) using the AD algorithm
Step 2.3: Send the results of **Step 2.2** to the BFGS optimizer to solve decision variables, $x_{odp(\bar{m})}^{(k)}$and $c_{od(\bar{m})}^{d(k)}$
For each component
Find the optimal variable $x_{odp(\bar{m})}^{(k)}$
Find the optimal variable $c_{od(\bar{m})}^{d(k)}$
Compute the decision variables shown in Eqs. (4.11a) – (4.11g)
End for
Step 3: Compute the defined constraints
Step 3.1: Compute the primal problem, Eq. (4.8a) with $x_{odp(\bar{m})}^{(k)}$
Step 3.2: Compute the constraints, Eq. (4.8d) and Eq. (4.5) with the optimal variables $x_{odp(\bar{m})}^{(k)}$ and $c_{od(\bar{m})}^{d(k)}$
Step 3.3: Compute the dualized optimization function, Eq. (4.10) with $x_{odp(\bar{m})}^{(k)}$ and $c_{od(\bar{m})}^{d(k)}$
Step 4: Evaluate the convergence criterion and constraints
Set up the feasibility tolerance gap ε_{gap} for BFGS (i.e., 1e-03)
Check the relative gap $\frac{\mathcal{L}_{CGAL}^{k+1} - \mathcal{L}_{CGAL}^{k}}{\mathcal{L}_{CGAL}^{k}} \times 100\%$ with ε_{gap}
Check the satisfaction of the defined conditions
Update the Lagrangian multipliers, Eq. (4.14) – (4.15) if constraints are not satisfied*

Note*: If the solution does not satisfy the criterion defined in **Step 4, then go back to **Step 2**.

Remark 2. In order to have computational easiness for the quadratic penalties, $\boldsymbol{\rho}$, we use constant values (small numbers).

Remark 3. We set up the initial values of $\boldsymbol{\lambda}$ as zero.

Remark 4. To generate the number of feasible paths and the travel time costs on those paths, we execute a traffic assignment package, DTALite published by Zhou and Taylor (2014).

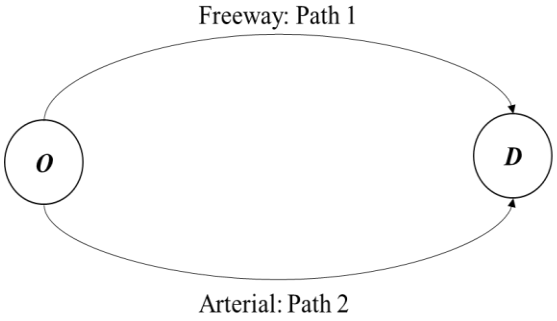
4.5. Numerical Experiments

In this section, two sets of numerical experiments are conducted to demonstrate the feasibility of finding optimal solutions through the proposed computational graph-based optimization program and theoretically linking transportation demand and network models. The first experiment aims to prove the accuracy of the solutions obtained by the gradient-based algorithm. The second experiment evaluates the performance of handling a real road network (Chicago skeleton network). Furthermore, we confirm that the algorithm can find network equilibrium states while fully linking travel demand models and network assignment models. Please note that in order to create the initial values of path flows and path travel times, we utilize the column generation-based DTALite package published by Zhou and Taylor (2014) and assume total demand in each origin node is given.

4.5.1. A Small Network (Iterative Solution, MSA, and Gradient Descent Algorithm)

This subsection examines the computational accuracy of the gradient-based optimization, compared to an iterative method and method of successive averages (MSA). Specifically, comparing reference values (path flows and travel costs) derived by an analytical formulation, we first check the performance accuracy of our approach and the number of

iterative steps to find optimal values. The simple corridor network includes two links (or paths) connecting one OD pair. The network attributes of the given corridor are described in Figure 4.3, and Eq. (4.1b) is used to catch travel mode choice behaviors.



Configuration	Freeway	Arterial
$FFTT_a$	20 min	30 min
CAP_a	4000 veh/h	3000 veh/h
TD_o	11692.2 Trips	
BPR Function (α)	1.0	
BPR Function (β)	1.0	

Figure 4.3. Small Network Illustrating the Configurations of Freeway and Arterial (Li et al., 2017)

In order to compute the true reference values, the analytical formulation of identifying user equilibrium (UE) conditions between two paths is derived. To readily derive the analytic expression, parameter values of α and β in the BPR function are assigned as 1. Then, Eq. (4.6b) can be expressed as: $\tau_a = FFTT_a\left(1 + (v_a/\text{CAP}_a)\right), \forall a$ where v_a can be expressed as $x_{odp(\bar{m})}$, and the other path flow variable can be written as $f_{od(\bar{m})} - x_{odp(\bar{m})}$ through the reducing gradient algorithm. By considering the UE condition, $\tau_a^F = \tau_a^A$, decision variables $x_{odp(\bar{m})}$ are calculated, and travel time costs $c_{od(\bar{m})}$ are also obtained.

$$FFTT_a^F\left(1 + \frac{x_{odp(\bar{m})}}{\text{CAP}_a^F}\right) = FFTT_a^F\left(1 + \frac{(f_{od(\bar{m})} - x_{odp(\bar{m})})}{\text{CAP}_a^A}\right) \tag{4.16}$$

Solving Eq. (4.16), we can derive path flows $x_{odp(\bar{m})}$ to find UE, which is 5333.33 vehicles/hour, and travel time cost $c_{od(\bar{m})}$ is 46.67 minutes computed by the first order BPR function. With these true reference values, we check the convergence and the performance

accuracy of each benchmark method. To manage the common variable needed in both demand and supply, we split $c_{od(\bar{m})}$ into $c^d_{od(\bar{m})}$ and $c^s_{od(\bar{m})}$. With the gradient descent method, we compute $c^d_{od(\bar{m})}$, and $c^s_{od(\bar{m})}$. On the other hand, the benchmark follows the heuristic algorithm:

Iterative method:

$$c^d_{od(\bar{m})}(k+1) = c^s_{od(\bar{m})}(k) \tag{4.17}$$

MSA method:

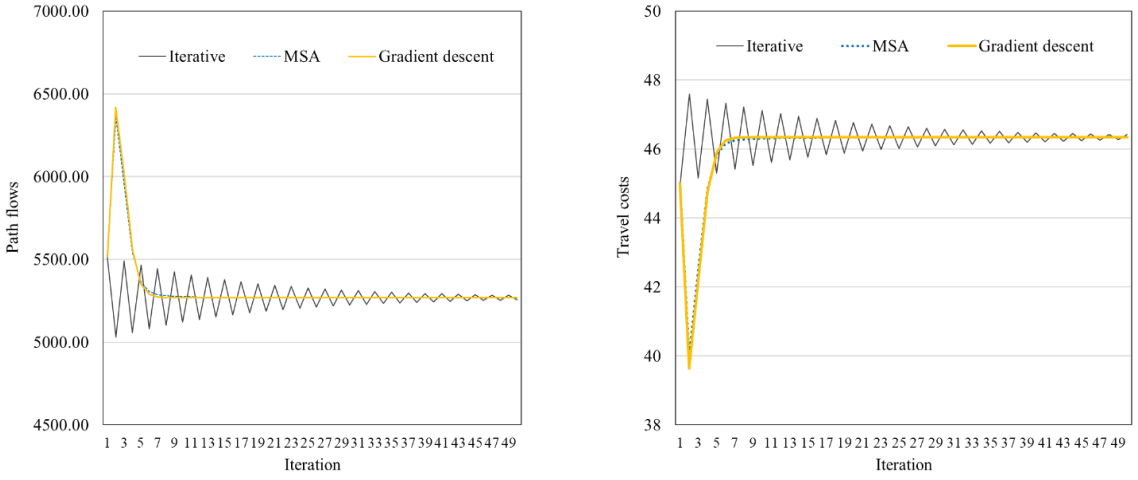
$$c^d_{od(\bar{m})}(k+1) = \frac{k}{k+1} c^d_{od(\bar{m})}(k) + \frac{1}{k+1} c^s_{od(\bar{m})}(k) \tag{4.18}$$

Analytical gradient descent method:

$$c^d_{od(\bar{m})}(k+1) = c^d_{od(\bar{m})}(k) - \alpha \nabla F\left(c^d_{od(\bar{m})}(k)\right) \tag{4.19}$$

where $\nabla F\left(c^{d(k)}_{od(\bar{m})}\right)$ is the gradient of the defined constraint function and $F\left(c^{d(k)}_{od(\bar{m})}\right)$ is defined as $\frac{1}{2}\left(c^{d(k)}_{od(\bar{m})} - c^{s(k)}_{od(\bar{m})}\right)^2$, enhancing the convexity of the given equation. Based on an arbitrary constant value (i.e., training parameter α) and the gradient: $\nabla F\left(c^{d(k)}_{od(\bar{m})}\right) = \left(c_d - c_s(c_d)\right) \times \left(1 - TD_o \times \left(\frac{FFTT^F_a}{3\mathrm{CAP}^F_a} + \frac{FFTT^A_a}{2\mathrm{CAP}^A_a} - \frac{FFTT^A_a}{3\mathrm{CAP}^A_a}\right) \times \left(-\frac{\theta e^{\theta(c_d + c_t)}}{\left(e^{\theta c_d} + e^{\theta c_T}\right)^2}\right)\right)$, Eq. (4.19) can be updated. With $c^d_{od(\bar{m})}(k)$, $c^s_{od(\bar{m})}(k)$, Eq. (4.1b), and Eq. (4.6b), composite values are computed. In particular, $f_{od(\bar{m})}(k+1)$ is derived by Eq. (4.1b) with $c^d_{od(\bar{m})}(k)$, $x_{odp(\bar{m})}(k+1)$ is given by Eq. (1) with $f_{od(\bar{m})}(k+1)$, and the first order BPR function with $x_{odp(\bar{m})}(k+1)$ calculates $\tau^F_a(k+1)$ and $\tau^A_a(k+1)$ values. Lastly, computing the average of their path travel time, we can determine the travel cost $c^s_{od(\bar{m})}(k+1)$.

Figure 4.4 represents the convergence test between the three experiments. First, the iterative method shows the fluctuation in finding the solutions, and after 100 iterations, it derives the converged results, 5267.95 vehicles/hour and 46.34 minutes. On the one hand, the gradient descent method displays the similar shape in terms of finding optimal solutions, compared to MSA. The computed variables are 5268.75 vehicles/hour and 46.34 minutes, which are almost identical to the results of MSA, demonstrating the gradient-based methodology can find optimal solutions in different cases. It should be noted that due to the exemplified network and the linearly defined UE condition, we are able to establish analytical formulations which is not easy to show the effectiveness of the gradient-based algorithm. If link performance functions are defined as nonlinear functions and available paths increase, network problem can be extremely complicated (Saitz, 1999). Although this experiment demonstrates the major advantage of MSA is that it can avoid zigzagging displacements, and it does not reply on expensive line search operations such as Zhou et al. (2009) or Ryu et al. (2017), there are critical limitations in finding optimal path flow patterns and OD demand in a higher congestion level as well as a larger transportation network. Sbayti et al. (2007) stated that the MSA method, specifically applied in path-based formulations for medium-size networks, results in explicit storage of the path sets and path assignments, thereby increasing the computing iterations to find converged variables. Furthermore, due to the absence of a mathematical formulation for minimizing relative gap of travel costs between inferior paths and current optimal (auxiliary) paths, MSA has low reliability of the convergence, at high congestion conditions.



(a) Computation of $x_{odp(\bar{m})}$ (b) Computation of $c^{d}_{od(\bar{m})}$

Figure 4.4. Convergence Comparison Test (Iterative Method, MSA, and Gradient Descent Algorithm

4.5.2. Real-World Case Study (Chicago Transportation Network)

This subsection addresses the applicability of the computational graph-based optimization program in the Chicago transportation network, one of the highest transit ridership areas in the U.S. (Hughes-Cromwick and Dickens, 2018). Recognizing the importance of analyzing the impact of mode choice behaviors between personal vehicles and transit on the built network, we examine the capability of linking the behavior patterns and the network models for the temporal equilibrium while finding the optimal solutions for the flow equilibrium. According to Figure 4.5, the number of nodes and links is 933 and 2,950, respectively. In addition, under the congested traffic state, the number of generated paths is 285,959, and the number of origin-destination pairs is 142,890. That is, the total number of path flow variables $x_{odp(\bar{m})}$ and travel costs $c^{d}_{od(\bar{m})}$ to link the demand model and the network model is 428,849, and the size of a path-link incidence matrix is 285,959×2,950. In this paper, to evaluate the feasibility of the proposed mathematical optimization program and avoid the computational burden of finding a large set of decision variables, we select

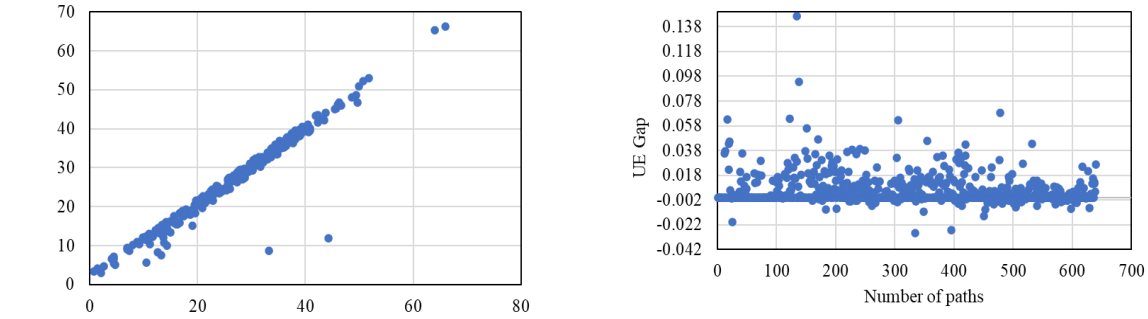
a few pairs of origin-destination matrices. Because of this, the number of $x_{odp(\bar{m})}$ to be computed is 3,835, and the number of $c^{d}_{od(\bar{m})}$ is 1,012 shown in Figure 4.5; the colored blue lines indicate path flow sequences. The dataset of this network can be downloaded from the website: https://github.com/asu-trans-ai-lab/Path4GMNS/tree/master/dataset.

Figure 4.5. Chicago Skeleton Transportation Network

In this paper, five different origin nodes are selected, and we sequentially find optimal solutions of each origin-destination pair using computational graph-based augmented Lagrangian relaxation (CGLR) method. To validate this optimization algorithm, we test the convergence of the objective function defined in Eq. (4.10) and the satisfaction of defined constraints in Eq. (4.5), Eq. (4.7a), and Eq. (4.8d). In other words, the three

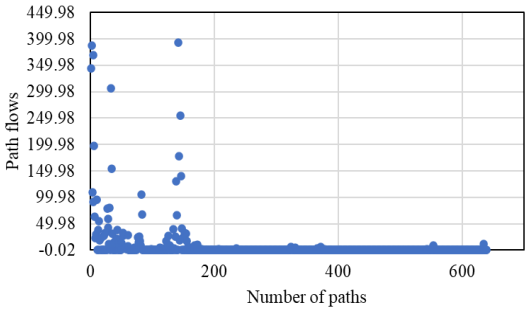
components of minimizing the optimization problem are testified: the bridge gap, which assures the linkage constraint of demand-supply units, the user equilibrium gap, which presents network equilibrium states, and the path flow positivity.

In Figure 4.6, each subplot indicates results of the defined constraints and the convergence of the objective function. We first illustrate the result from the origin id 1 with multiple destination ids; the number of *od* pairs is 229, and paths are 639. Figure 4.6-(a) shows the relative gap between demand and supply. If the optimized variables (i.e., $c^{d}_{od(\bar{m})}$ and $c^{s}_{od(\bar{m})}$) are not closely equal to each other, then the variance of the two variables is largely distributed. In this case, except for a few pairs, the proposed model is able to find the consistency solutions. Figure 4.6-(b) represents the user equilibrium condition where most of the measured data points are found near to zero, displaying the equilibrium state. The average of the UE gap in this given scenario is 0.058. Furthermore, Figure 4.6-(c) demonstrates the positivity of path flows $x_{odp(\bar{m})}$. Lastly, the convergence of the objective function is displayed in Figure 4.6-(d). The number of iterations of the BFGS algorithm is 500, and we set the tolerance value as 0.001. That is, if the relative gap between previous step and current step is smaller than the tolerance, then we stop the numerical optimizer. Checking the satisfaction of the restraints, we decide whether or not to update Lagrangian multipliers.

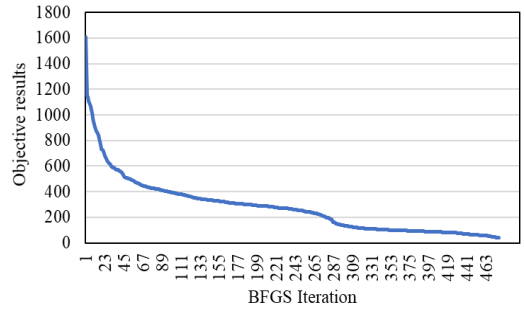


(a) Demand-supply gap comparison

(b) UE gap



(c) Path flow positivity



(d) Objective results

Figure 4.6. CGLR Results of Origin Node 1.

Assuming mutually exclusivity of each origin-destination pair, we execute numerical experiments of different OD pairs. The configuration setting for the BFGS algorithm follows the aforementioned values, and the average computing time of solving the objective function and obtaining optimal solutions is around 80 seconds. The average values of each constraint, namely the UE gap, the bridge gap, and the path flow positivity are provided in Table 4.3.

Table 4.3. Optimization Results: Five Different Origin Nodes

IDs	OD Pairs (Paths)	Time	User Equilibrium Gap			Path Flow Positivity			Demand-Supply Consistence gap		
			Min	Max	Avg	Min	Max	Avg	Min	Max	Avg
1	229 (639)	75 s	-0.04	11.02	0.06	-0.02	392.71	7.53	-2.79	32.49	-0.00
2	236 (763)	81 s	-0.12	48.48	0.17	-0.03	731.93	8.66	-4.20	10.18	0.00
3	247 (730)	80 s	-0.94	8.30	0.04	-0.16	1422.13	13.07	-3.71	55.18	0.00
4	250 (902)	83 s	-1.03	9.68	0.06	-0.17	990.58	9.94	-10.9	61.75	0.00
5	279 (801)	77 s	-0.48	22.98	0.08	-0.08	3033.68	20.44	-13.6	36.0	-0.00

4.6. Summary and Conclusions

In this study, by developing the analytical/gradient tractable structures with computational graphs, we proposed the augmented Lagrangian relaxation framework to find the primal and dual solutions for the consistent integration of demand-supply models and for the network equilibrium states. We first constructed a variational inequality formulation and constraints to follow the Wardrop's principle. By splitting the common variable used in both demand side and supply side, we created the linking constraints that can minimize the gap of demand-supply components. Furthermore, the augmented Lagrangian relaxation method was introduced to dualize the defined constraints, deriving the unconstrained minimization problem. To compute gradients of decision variables (i.e., optimal path flows and traveling costs) of the objective function, automatic differentiation (AD), specifically the reverse-mode AD algorithm, was conducted. Transmitting the gradients into the numerical optimizer (BFGS), we found the optimal decision variables, and the program keeps updating the Lagrangian multipliers to minimize the UE gap and the bridge gap

(demand-supply consistence gap) while maintaining path flows positivity. To demonstrate the effectiveness of the proposed approach, numerical experiments were conducted, including a small network and Chicago skeleton transportation network. The measured results showed that the computational graph-based augmented Lagrangian relaxation (CGLR) method can find the optimal path flow patterns and decoupled variables (generalized costs and traveling costs) as shown in Figure 4.6 and Table 4.3.

Along with the numerical results given by CGLR, this research was able to validate the capability of computing the analytical gradients of the decision variables defined in the proposed mathematical modeling structure and deriving consistent level-of-service and flow values of the modal split and traffic assignment steps in the new mathematical formulation. Compared to prior study in Table 4.4, our proposed method including the consistent coupling constraint would be an ideal framework that can be as simple as possible while sufficient descriptive ability for interactive transportation systems, answering pertinent questions asked by transportation decision-makers.

Table 4.4. Formulation for Interactive Variables (i.e., Costs in Demand Side and Supply Side)

Solution algorithm	**Formulation**	**Step size**	**Publication**
Method of successive average	$c_{k+1}^{d} = \frac{k-1}{k} c_k^d + \frac{1}{k} c_k^s$	Iterative steps $(1/k)$	Florian et al. (2002), Lin et al. (2008), Zhou (2008)
Constant weight	$c_{k+1}^{d} = (1-w)c_k^d + wc_k^s$	Fixed (w)	Boyce et al. (2008)
Gradient projection	$c_k^d = c_k^s(x_k) + \frac{1}{\beta_r}\ln(x_k)$	Self-adaptive	Zhou et al. (2009)
Descent direction	$c_k^d = t_k\big(x\big(f_k(\cdot)\big), \delta\big)$	Line search	Ryu et al. (2017)

Analytical gradients using AD	$c_k^d = c_k^s$	Lagrangian penalty	This paper

That is, the coupling formulations, presented by previous literatures in Table 4.1, are not modelled explicitly as constraints (e.g., gradient projection or descent direction). Instead, they are indirectly enforced through the optimality conditions at the final stage, resulting in the insufficiency of establishing highly interactive transportation systems. In comparison, our proposed mathematic programming model specifically includes and further dualizes the coupling condition, to better measure the demand and supply interactions along the solution-finding process.

Overall, although transportation modelers have developed comprehensive modeling structures with heuristic algorithms and followed iterative sequential processes to solve mathematical objective functions, finding optimal path flows and travel costs to satisfy equilibrium states, our methodology advances the kernel of coupling/developing interactive transportation systems: (1) a consistent mechanism to mathematically integrate the generalized accessibility cost on the demand side and the travel cost computed by traffic assignment, (2) gradient-oriented approximation to solve complex objective functions composed of the costs, path flows, OD flow patterns, and choice probabilities, and lastly (3) the computational power of calibrating a large set of the decision variables in the real transportation network. In other words, determining step sizes on the basis of the analytical gradients could be numerically stable fashion, compared to "averaging" manner. Furthermore, the analytically modeled structure in terms of flow, cost, and choice probability would mitigate inconsistency between demand and supply variables. Lastly,

the solution algorithms designed to directly minimize and reduce the model gap measure iteratively could be alternative ways to meet the requirements in solving consistent integrated travel demand and network modelling architectures.

With this graphically-oriented mathematical modeling framework and its solution algorithm using automatic differentiation (AD), our research will further focus on the two aspects:

- Extension of the modal split-traffic assignment modeling structure to consider more realistic scenarios (behavioral richness and spatio-temporal dimension): As stated in Zhou et al. (2009) and Ryu et al. (2017), in order to have more behavioral richness in a top-down sequential system, a hierarchical choice structure will be defined by satisfaction functions or log-sum terms and be constructed by computational graphs to calculate the analytical gradients with respect to decision variables. By doing so, we would generate more behavioral realistic coupling constraints in the model while ensuring the computational efficiency in dealing with the highly complex composite terminologies. Furthermore, for consistency in the representation of behavioral units, spatial relationships, and temporal scales (Pendyala et al., 2012), spatial-temporal constraints are to be dualized in a potential modeling framework.
- Dynamic programming (DP) for a larger scale network experiment: With high-performance computing techniques used for the calibration of a large set of decision variables, we will evaluable the applicability of a programming architecture in national-level scales, capturing a wide range of emerging patterns in transportation

ecosystem (Zhou, 2021). To improve the computing efficiency of finding optimal flow patterns, costs, and decision-maker's choice probability while satisfying consistency in the representation listed above, we would study different types of quasi-newton methods such as limited-BFGS or a first-order method, the Adam optimization algorithm (Kingma and Ba, 2014). In consequence, we can further deal with more complex transportation problems, congestion, emergency response planning, or toll/incentive issues in built network environments.

Appendix A. The Fixed-Point Formulation of the Interaction between Transportation Demand and Supply

$$x = \text{feasible path flows obtained by Path4GMNS} \quad \text{(A.3)}$$

$$\tau(x,\alpha,\beta) = \text{link traveling cost} \quad \text{(A.2)}$$

$$t\big(\tau(x,\alpha,\beta)\big) = \text{path traveling cost} \quad \text{(A.3)}$$

$$\text{MNL}\big(t\big(\tau(x,\alpha,\beta)\big);\theta,U\big) = \text{choice behavior resulting from (A.3)} \quad \text{(A.4)}$$

$$f\big(\text{MNL}\big(t\big(\tau(x,\alpha,\beta)\big);\theta,U\big),\text{OD}\big) = \text{path flows based on OD demand and (A.4)} \quad \text{(A.5)}$$

$$\delta\Big(f\big(\text{MNL}\big(t\big(\tau(x,\alpha,\beta)\big);\theta,U\big),\text{OD}\big)\Big) = \text{path cost mapped by the incidence matrix} \quad \text{(A.6)}$$

$$t\big(\tau(x,\alpha,\beta)\big) = \delta\Big(f\big(\text{MNL}\big(t\big(\tau(x,\alpha,\beta)\big);\theta,U\big),\text{OD}\big)\Big) = \text{the fixed-point formulation } (x \geq 0 \text{ and } x \in D) \quad \text{(A.7)}$$

Appendix B. Difference between Automatic Differentiation and the Chain Rule in Computing Gradients of a Composite Function

The fundamental process of decomposing differentials is similar to the chain rule approach, but there are two different features that distinguish AD from the chain rule. As stated in Bartholomew et al. (2000), AD carries floating point numerical values, instead of differentiating the symbolic expressions decomposed and reduces the complexity of computing the complex composite function. Furthermore, using intermediate variables as checkpoints, the AD algorithm can save the memory required. For instance, we provide a

simple example, $h(x) = f\left(g\left(k(x)\right)\right)$ and show how the intermediate variables are defined. Defining intermediate variables, w_i, in each decomposed element, we can not only store the computed gradients but also carry numerical values. On the other hand, as the pure chain rule approach needs to compute the symbolic expressions without the stored variables (i.e., intermediate variables), the computational cost is much higher than AD.

Automatic Differentiation (AD)	*Chain Rule*
$\frac{\partial h(x)}{\partial x} = \frac{\partial f(w_2)}{\partial w_2}\frac{\partial w_2}{\partial w_1}\frac{\partial w_1}{\partial w_0}$ where, $w_0 = x$ $w_1 = k(w_0)$ $w_2 = g(w_1)$ $w_3 = f(w_2) = h(x)$	$\frac{\partial h(x)}{\partial x} = f'\left(g\left(k(x)\right)\right)\ g'\left(k(x)\right)k(x)$

CHAPTER 5

5. CONCLUSION

This chapter begins with a brief summary of the proposed methodologies and findings. Section 5.2 addresses the perspective on the contributions of this dissertation to the state of the art of computational graph-based frameworks. In addition, the following subsection discusses broader application of the innovative approaches in the transportation planning field.

5.1. Research Overview

With the development of three key components in the field of machine learning (ML), namely computing power, a large bulk of data sources, and new algorithms, planners and researchers have observed successful cases of analyzing multiple data sources efficiently, discovering unseen patterns in the big chunk of data, as well as enhancing predictability in transportation planning. Encouraged by the advantages of ML applications, this research explores the underlying algorithm behind ML: *computational graph* (CG) and *automatic differentiation* (AD), which enables defined mathematical models to be designed as graph-oriented structures and to find analytical gradient information, finding optimal solutions.

Leveraging the two key algorithms, we propose the promising guidance to theoretically and practically embed them into transportation planning methods and the further direction of how to extend the applicability of ML in the transportation community using three different frameworks developed. First, we present the hybrid modeling structure that can combine the regression model with the advanced recurrent neural network (also known as Long-Short Term Memory (LSTM)). Second, we reformulate the

discrete choice models using the computational graphs, enhance the computation power of estimating parametric variables based on AD, and address the modeling flexibility that can efficiently be incorporated with ML methods. Lastly, we develop the analytically-driven mathematical formulation to integrate travel demand and network supply model through the graph-oriented method and propose the analytical gradient information with respect to the proposed structure.

Overall, through the proposed frameworks, the advantages of applying graph-oriented planning models with AD are presented, we can potentially expect to see further benefits: The use of simple chain rules typically involves numerical computing of gradients, that is, perturbing each element of variables one by one. This type of finite difference method could be inefficient as it needs to perform a forward pass for each derivative. Additionally, the simple finite difference method might lead to infeasible solutions for large variable ranges, and it is typically subject to round-off errors. Most automatic differentiation software systems, including Autograd and TensorFlow, explicitly construct the computation graph. Thus, the modern automatic differentiation method can store the optimization algorithm symbolically, and its efficient implementation of the chain rule based on a dynamic programming principle can systematically carry out the forward and backward passes of Jacobian gradient vectors with nested operation. Furthermore, the integration of back propagation and computational graphs, through higher order of vectorization, can further improve the computational efficiency by providing a high-quality estimate of the deviation terms.

5.2. Research Contributions

This subsection presents research contributions for the advancement of transportation planning models. As acknowledged in transportation planning projects, conventional modeling approaches have difficulty of systematically capturing emergent behaviors and network flow patterns, achieving computational efficiency in realistic networks, and incorporating big data sources.

Based on the computational graph-based frameworks and machine learning architectures (e.g., deep neural networks), this dissertation proposes the next generation tool to enhance conventional transportation planning methods, eventually introducing a hub-system that seamlessly incorporate multiple data sources, travel demand, and network models. This research provides the following key contributions.

I. The hybrid modeling framework constructed by econometric models and deep learning techniques can not only interpret significant factors of potential trends but also sense heterogeneous patterns, enabling planners to manage dynamics of on-demand mobility services.

II. The computational graph-based discrete choice models can show higher computational performance in dealing with numerous behavioral parameters and yield the flexibility of extending the computational structures to broaden a set of accessibility-oriented planning applications.

III. The alternative mathematical formulation to integrate travel demand and network models using computational graphs can be capable of solving large-scale transportation networks and can be compatible with existing modeling

software for activity-based model (ABM) and dynamic traffic assignment (DTA), reducing programming expenses.

This graph-oriented representation successfully embeds the underlying algorithms of ML (CG and AD) into transportation planning methods, specifically modeling structures, accuracy and efficiency of model estimation, and analytical solutions for network optimization algorithms. More importantly, this dissertation systematically introduces the way of establishing the modeling language consistency that can simultaneously use the strengths of machine learning algorithms and the theory-driven models. Accordingly, transportation planners and developers can efficiently manage/analyze massive datasets such as a large set of household surveys, large-scale networks, and passively collected data sources (e.g., GPS or mobile phone), understanding interactive transportation systems shaped by travelers' behavioral patterns and transportation infrastructures in metropolitan areas. Furthermore, the computational graph-based models can have a significant potential to analytically extend different types of modeling structures that can discover causal inference relationships in a data-rich environment, helping planners to statistically interpret hidden patterns in the given datasets.

To further illustrate our overarching modeling approach, we use the conceptual framework in in Figure 5.1 to highlight the needed consistency of modeling language to build behavioral models and machine learning architectures. We hope this CG-oriented perspective could allow us to seamlessly integrate traditional econometric traveler behavior models with new and emerging data-driven approaches. Overall, the proposed graph-based modeling framework not only offers the flexibility of expanding conventional modeling

approaches but also enables planners and policy makers to estimate the system-wide utility more precisely for different projects and demand management alternatives, potentially leading to better decisions for improved transportation systems.

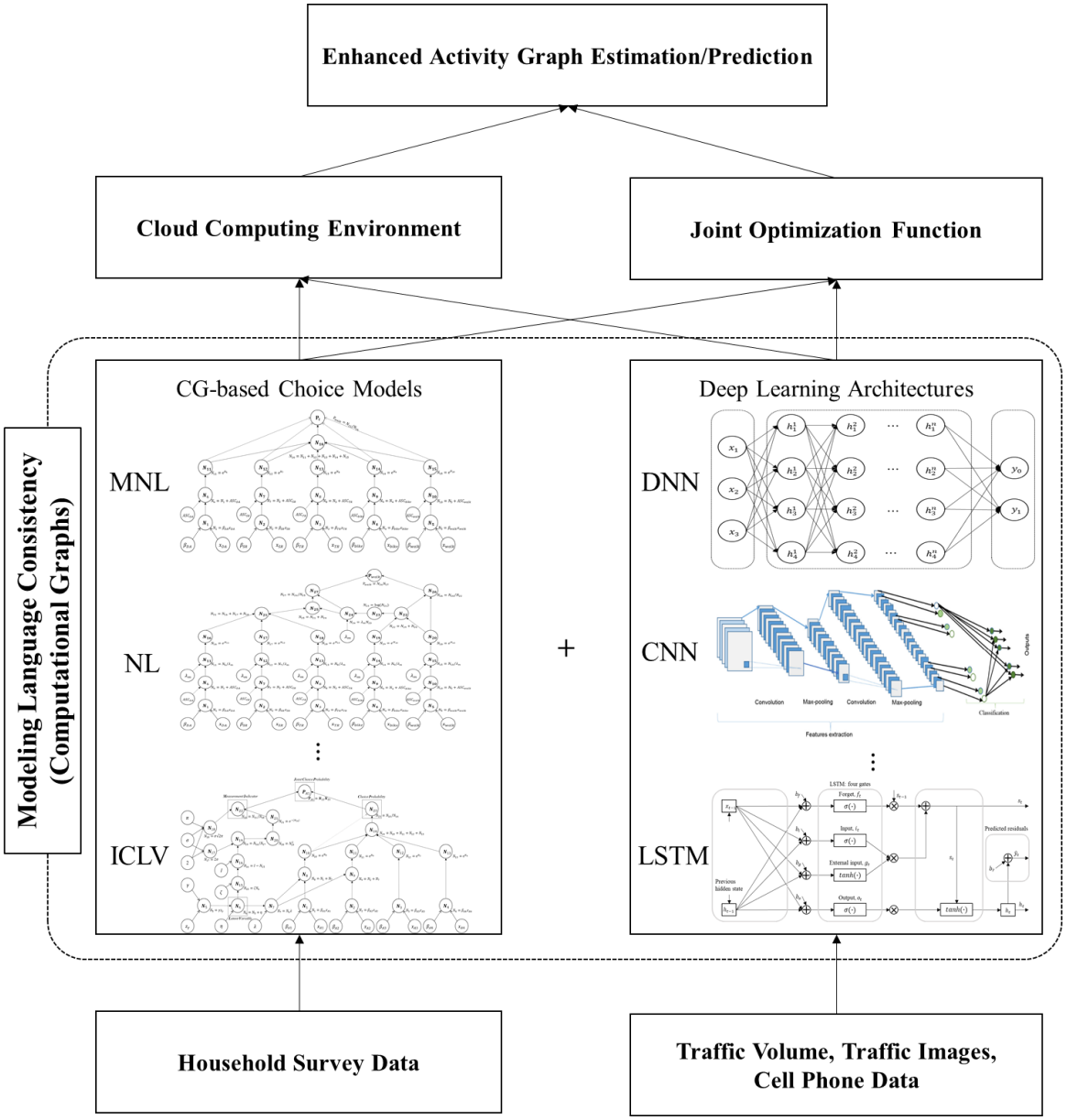


Figure 5.1. Illustration of Developing a Consistent Modeling Structure between Choice Models and Deep Learning (Using Examples from CNN in Alom et al. (2019) and LSTM in Kim et al. (2020))

5.3. Future Research

With advanced transportation planning models and the innovative algorithms, the graph-oriented frameworks will be applicable in transportation projects and help to answer research questions. Specifically, the developed methodologies can be promising potential in the following lists.

(1) Supporting active demand management (ADM) system in response to mobility services/technologies: in efforts to reduce annual cost of congestion in the U.S. and achieve sustainable highways, community leaders and transportation planners have adapted active demand management (ADM) which dynamically controls traffic states using information and technologies. Aiming to enhance the strategy for developing useful decision support systems, this research will continue to focus on hybrid modeling frameworks constructed by econometric models and machine learning architectures to efficiently integrate multiple data sources (e.g., traffic images and travelers' choices) while simultaneously interpreting significant factors of demand and predicting travel patterns. The combination of the statistical models and data-fitting models will help decision makers to properly quantify and balance traffic usage, establishing reliable transportation systems.

(2) Efficient estimation algorithms for measuring multi-modal accessibility in transportation: The development of new mobility technologies such as autonomous vehicles has the potential to reduce a travel-limiting disability that accomplishes daily activities. Particularly, people with disabilities can achieve the increased accessibility for their trips. In light of this, it is extremely imperative to develop a quantitative formulation to measure the accessibility with high performance computing. To establish theoretically reliable models that can model individuals' choices while efficiently handling the complex

composite structures, this framework will define graph-oriented formulations that combine utility-based methodologies and automatic differentiation (AD).

(3) Deployment of the integrated models in cloud computing (CC): Cloud Computing (CC) has represented a promising approach for real-time large-scale system modeling, ubiquitous communication, and diverse data synthesis by demonstrating the effectiveness in terms of cost and managing computers. In order to couple the technology, this project will implement open-standard and open-source based architecture, allowing them to be universally applicable in the domains of transportation planning, traffic analysis, network behavior, and demand forecasting. This connectivity can have a positive potential to enhance intelligent transportation system.

REFERENCES

Abadi, M., Barham, P., Chen, J., Chen, Z., Davis, A., Dean, J., Devin, M., Ghemawat, S., Irving, G., Isard, M. and Kudlur, M., 2016. Tensorflow: A system for large-scale machine learning. In *12th {USENIX} Symposium on Operating Systems Design and Implementation ({OSDI} 16)* (pp. 265-283). Software available from https://www.tensorflow.org

Agrawal, A., Verschueren, R., Diamond, S. and Boyd, S., 2018. A rewriting system for convex optimization problems. Journal of Control and Decision, *5*(1), pp.42-60.

Al-Maqaleh, B.M., Al-Mansoub, A.A. and Al-Badani, F.N., 2016. Forecasting using artificial neural network and

Alom, M.Z., Taha, T.M., Yakopcic, C., Westberg, S., Sidike, P., Nasrin, M.S., Hasan, M., Van Essen, B.C., Awwal, A.A. and Asari, V.K., 2019. A state-of-the-art survey on deep learning theory and architectures. Electronics, 8(3), p.292.

Altman, E.I., Marco, G. and Varetto, F., 1994. Corporate distress diagnosis: Comparisons using linear discriminant analysis and neural networks (the Italian experience). Journal of banking & finance, 18(3), pp.505-529.

Ashraf, J., Bakhshi, A. D., Moustafa, N., Khurshid, H., Javed, A., & Beheshti, A., 2020. Novel Deep Learning-Enabled LSTM Autoencoder Architecture for Discovering Anomalous Events From Intelligent Transportation Systems. IEEE Transactions on Intelligent Transportation Systems.

Bartholomew-Biggs, M., Brown, S., Christianson, B. and Dixon, L., 2000. Automatic differentiation of algorithms. Journal of Computational and Applied Mathematics, 124(1-2), pp.171-190.

Bastien, F., Lamblin, P., Pascanu, R., Bergstra, J., Goodfellow, I., Bergeron, A., Bouchard, N., Warde-Farley, D. and Bengio, Y., 2012. Theano: new features and speed improvements. arXiv preprint arXiv:1211.5590.

Baydin, A.G., Pearlmutter, B.A., Radul, A.A. and Siskind, J.M., 2017. Automatic differentiation in machine learning: a survey. The Journal of Machine Learning Research, 18(1), pp.5595-5637.

Beck, T. and Fischer, H., 1994. The if-problem in automatic differentiation. Journal of Computational and Applied Mathematics, 50(1-3), pp.119-131.

Belezamo, B., 2020. Data-driven Methods for Characterizing Transportation System Performances Under Congested Conditions: A Phoenix Study (Doctoral dissertation, Arizona State University).

Ben-Akiva, M., Walker, J., Bernardino, A.T., Gopinath, D.A., Morikawa, T. and Polydoropoulou, A., 2002. Integration of choice and latent variable models. Perpetual motion: Travel behaviour research opportunities and application challenges, pp.431-470.

Bhavsar, P., Safro, I., Bouaynaya, N., Polikar, R. and Dera, D., 2017. Machine learning in transportation data analytics. In Data analytics for intelligent transportation systems (pp. 283-307). Elsevier.

Bierlaire, M., 2003. BIOGEME: A free package for the estimation of discrete choice models. In Swiss Transport Research Conference (No. CONF).

Bierlaire, M., 2016. PythonBiogeme: a short introduction (Report TRANSP-OR 160706, Ecole Polytechnique F´ed´erale de Lausanne).

Bottou, L., 2010. Large-scale machine learning with stochastic gradient descent. In Proceedings of COMPSTAT'2010 (pp. 177-186). Physica-Verlag HD.

Bowman, J.L., 1998. The day activity schedule approach to travel demand analysis (Doctoral dissertation, Massachusetts Institute of Technology).

Box, G.E., Jenkins, G.M., Reinsel, G.C. and Ljung, G.M., 2015. Time series analysis: forecasting and control. John Wiley & Sons.

Boyce, D., O'Neill, C.R. and Scherr, W., 2008. Solving the sequential travel forecasting procedure with feedback. Transportation Research Record, 2077(1), pp.129-135.

Boyce, D., Ralevic-Dekic, B. and Bar-Gera, H., 2004. Convergence of traffic assignments: how much is enough?. Journal of Transportation Engineering, 130(1), pp.49-55.

Brathwaite, T. and Walker, J.L., 2018a. Asymmetric, closed-form, finite-parameter models of multinomial choice. Journal of choice modelling, 29, pp.78-112.

Brathwaite, T., & Walker, J. L., 2018b. Causal inference in travel demand modeling (and the lack thereof). Journal of choice modelling, 26, 1-18.

Brathwaite, T., 2017. GitHub repository, https://github.com/timothyb0912/pylogit/

Cantarella, G.E., Cartenì, A. and de Luca, S., 2015. Stochastic equilibrium assignment with variable demand: theoretical and implementation issues. *European Journal of Operational Research*, *241*(2), pp.330-347.

Chang, X., Wu, J., Liu, H., Yan, X., Sun, H. and Qu, Y., 2019. Travel mode choice: a data fusion model using machine learning methods and evidence from travel diary survey data. Transportmetrica A: Transport Science, 15(2), pp.1587-1612.

Chapra, S.C. and Canale, R.P., 2011. Numerical methods for engineers (Vol. 1221). New York: Mcgraw-hill.

Chen, A., Lee, D.H. and Jayakrishnan, R., 2002. Computational study of state-of-the-art path-based traffic assignment algorithms. Mathematics and computers in simulation, 59(6), pp.509-518.

Chen, B.Y. and Kwan, M.P., 2020. Special Issue on Spatiotemporal Big Data Analytics for Transportation Applications, Transportmetrica A: Transport Science, 16:1, 1-4.

Chen, C., Ma, J., Susilo, Y., Liu, Y. and Wang, M., 2016. The promises of big data and small data for travel behavior (aka human mobility) analysis. Transportation research part C: emerging technologies, 68, pp.285-299.

Chen, X.M., Zahiri, M. and Zhang, S., 2017. Understanding ridesplitting behavior of on-demand ride services: An ensemble learning approach. Transportation Research Part C: Emerging Technologies, 76, pp.51-70.

Chu, Y.L., 2018. Implementation of a new network equilibrium model of travel choices. Journal of Traffic and Transportation Engineering (English Edition), 5(2), pp.105-115. Computational and Applied Mathematics, 50(1-3), pp.119-131.

Cui, Yu, Chuishi Meng, Qing He, and Jing Gao., 2018. Forecasting current and next trip purpose with social media data and Google Places. Transportation Research Part C: Emerging Technologies 97: 159-174.

Esser, J. and Nagel, K., 2001. Iterative demand generation for transportation simulations. The leading edge of travel behavior research, pp.659-681.

FHWA. 2012. ATDM Program Brief: An Introduction to Active Transportation and Demand Management. Publication Number: FHWA-HOP-12-032, U.S. Department of Transportation, Federal Highway Administration (FHWA), Washington, DC. 2 p.

Fischer, B., 2015. In Uber vs. taxi cab fight, expense reports offer telling barometer. The Business Journals. Retrieved April, 2, p.2017.

Fisher, M.L., Jörnsten, K.O. and Madsen, O.B., 1997. Vehicle routing with time windows: Two optimization algorithms. Operations research, 45(3), pp.488-492.

Florian, M., Wu, J.H. and He, S., 2002. A multi-class multi-mode variable demand network equilibrium model with hierarchical logit structures. In Transportation and network analysis: Current trends (pp. 119-133). Springer, Boston, MA.

Fotheringham, A.S., 1986. Modelling hierarchical destination choice. Environment and planning A, 18(3), pp.401-418.

Gerte, R., Konduri, K.C. and Eluru, N., 2018. Is there a limit to adoption of dynamic ridesharing systems? evidence from analysis of uber demand data from new york city. Transportation Research Record, 2672(42), pp.127-136.

Goel, H., Melnyk, I. and Banerjee, A., 2017. R2N2: Residual recurrent neural networks for multivariate time series forecasting. arXiv preprint arXiv:1709.03159.

Golshani, N., Shabanpour, R., Mahmoudifard, S.M., Derrible, S. and Mohammadian, A., 2018. Modeling travel mode and timing decisions: Comparison of artificial neural networks and copula-based joint model. Travel Behaviour and Society, 10, pp.21-32.

Goodfellow, I., Bengio, Y. and Courville, A., 2016. Sequence Modeling: Recurrent and Recursive Nets. Deep Learning, pp.367-415.

Griewank, A. and Walther, A., 2008. Evaluating derivatives: principles and techniques of algorithmic differentiation. Society for Industrial and Applied Mathematics.

Gu, Y., Qian, Z.S. and Chen, F., 2016. From Twitter to detector: Real-time traffic incident detection using social media data. Transportation research part C: emerging technologies, *67*, pp.321-342.

Guignard, M., 2003. Lagrangean relaxation. Top, 11(2), pp.151-200.

Gunning, D. (2017). Explainable artificial intelligence (xai). Defense Advanced Research Projects Agency (DARPA), nd Web, 2.

Guo, J., C. Brakewood., C. Liu., and K. Won., 2018. Analysis and comparison of Uber, Taxi, and Uber request via Transit.

Han, Y., Zegras, C., Pereira, F.C. and Ben-Akiva, M., 2020. A Neural-embedded Choice Model: TasteNet-MNL Modeling Taste Heterogeneity with Flexibility and Interpretability. arXiv preprint arXiv:2002.00922.

Hao, J.Y., Hatzopoulou, M. and Miller, E.J., 2010. Integrating an activity-based travel demand model with dynamic traffic assignment and emission models: Implementation in the Greater Toronto, Canada, area. Transportation Research Record, 2176(1), pp.1-13.

Harwood, L., 2020. Shared Micromobility Policies, Permits, and Practices – NCHRP Synthesis 20-05/Topic 52-13

Hashem, I.A.T., Chang, V., Anuar, N.B., Adewole, K., Yaqoob, I., Gani, A., Ahmed, E. and Chiroma, H., 2016. The role of big data in smart city. International Journal of Information Management, 36(5), pp.748-758.

Hasnat, M.M. and Hasan, S., 2018. Identifying tourists and analyzing spatial patterns of their destinations from location-based social media data. Transportation Research Part C: Emerging Technologies, 96, pp.38-54.

He, F. and Shen, Z.J.M., 2015. Modeling taxi services with smartphone-based e-hailing applications. Transportation Research Part C: Emerging Technologies, 58, pp.93-106.

Hess, S. and Palma, D., 2019. Apollo: A flexible, powerful and customisable freeware package for choice model estimation and application. Journal of choice modelling, 32, p.100170.

Hochreiter, S. and Schmidhuber, J., 1997. Long short-term memory. Neural computation, 9(8), pp.1735-1780.

Hughes-Cromwick, M. and Dickens, M., 2018. APTA 2017 Public Transportation Fact Book. Integrating Econometric Models and Machine Learning Algorithms in Emerging Data-Driven Analytical Environments. Transportmetrica A: Transport Science, pp.1-35.

Jayakrishnan, R., Tsai, W.T., Prashker, J.N. and Rajadhyaksha, S., 1994. A faster path-based algorithm for traffic assignment.

Kamga, C., Yazici, M.A. and Singhal, A., 2015. Analysis of taxi demand and supply in New York City: implications of recent taxi regulations. Transportation Planning and Technology, 38(6), pp.601-625.

Karlaftis, M.G. and Vlahogianni, E.I., 2011. Statistical methods versus neural networks in transportation research: Differences, similarities and some insights. Transportation Research Part C: Emerging Technologies, 19(3), pp.387-399.

Ke, J., Zheng, H., Yang, H. and Chen, X.M., 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, pp.591-608.

Kelley, Jr, J. E., 1960. The cutting-plane method for solving convex programs. Journal of the society for Industrial and Applied Mathematics, 8(4), 703-712.

Kim, J., Rasouli, S. and Timmermans, H., 2016. A hybrid choice model with a nonlinear utility function and bounded distribution for latent variables: application to purchase intention decisions of electric cars. Transportmetrica A: Transport Science, 12(10), pp.909-932.

Kim, T., Sharda, S., Zhou, X. and Pendyala, R.M., 2020. A stepwise interpretable machine learning framework using linear regression (LR) and long short-term memory (LSTM): City-wide demand-side prediction of yellow taxi and for-hire vehicle

(FHV) service. Transportation Research Part C: Emerging Technologies, 120, p.102786.

Kim, T., Zhou, X. and Pendyala, R.M., 2021. Computational Graph-based Framework for Integrating Econometric Models and Machine Learning Algorithms in Emerging Data-Driven Analytical Environments. Transportmetrica A: Transport Science, (just-accepted), pp.1-35.

Kim, T., Zhou, X. and Pendyala, R.M., 2021. GitHub repository, https://github.com/Taehooie/CGChoice

Kingma, D.P. and Ba, J., 2014. Adam: A method for stochastic optimization. arXiv preprint arXiv:1412.6980.

Koppelman, F.S. and Bhat, C., 2006. A self instructing course in mode choice modeling: multinomial and nested logit models.

Kuflik, T., Minkov, E., Nocera, S., Grant-Muller, S., Gal-Tzur, A. and Shoor, I., 2017. Automating a framework to extract and analyse transport related social media content: The potential and the challenges. Transportation Research Part C: Emerging Technologies 77: 275-291.

Kumar, R., Aggarwal, R.K. and Sharma, J.D., 2015. Comparison of regression and artificial neural network models for estimation of global solar radiations. Renewable and Sustainable Energy Reviews, 52, pp.1294-1299.

Lam, C. and Liu, M., 2019. Toward Inclusive Mobility: Ridesharing Mitigates Geographical Disparity in Transportation. Available at SSRN 2997190

Lam, W.H. and Huang, H.J., 2003. Combined activity/travel choice models: time-dependent and dynamic versions. Networks and Spatial Economics, 3(3), pp.323-347.

Laptev, N., Yosinski, J., Li, L.E. and Smyl, S., 2017. Time-series extreme event forecasting with neural networks at uber. In International Conference on Machine Learning (No. 34, pp. 15).

Lavieri, P.S., Dias, F.F., Juri, N.R., Kuhr, J. and Bhat, C.R., 2018. A model of ride-sourcing demand generation and distribution. Transportation Research Record, p.0361198118756628.

LeBeau, P., 2020. Waymo launches delivery service after raising $2.25 billion. CNBC. Retrieved from https://www.cnbc.com/2020/03/02/waymo-launches-delivery-service-after-raising-2point25-billion.html

LeCun, Y. and Bengio, Y., 1995. Convolutional networks for images, speech, and time series. The handbook of brain theory and neural networks, 3361(10), p.1995.

LeCun, Y., Bengio, Y. and Hinton, G., 2015. Deep learning. nature, 521(7553), pp.436-444.

Li, M., Rouphail, N. M., Mahmoudi, M., Liu, J., & Zhou, X., 2017. Multi-scenario optimization approach for assessing the impacts of advanced traffic information under realistic stochastic capacity distributions. Transportation Research Part C: Emerging Technologies, 77, 113-133.

Lin, D.Y., Eluru, N., Waller, S.T. and Bhat, C.R., 2008. Integration of activity-based modeling and dynamic traffic assignment. Transportation Research Record, 2076(1), pp.52-61.

Lin, K., Zhao, R., Xu, Z., & Zhou, J., 2018. Efficient large-scale fleet management via multi-agent deep reinforcement learning. In Proceedings of the 24th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining (pp. 1774-1783).

Lipton, Z.C., 2018. The mythos of model interpretability. Queue, 16(3), pp.31-57.

Lo, H.K. and Chen, A., 2000. Reformulating the traffic equilibrium problem via a smooth gap function. Mathematical and computer modelling, 31(2-3), pp.179-195.

Lu, C.C., Mahmassani, H.S. and Zhou, X., 2009. Equivalent gap function-based reformulation and solution algorithm for the dynamic user equilibrium problem. Transportation Research Part B: Methodological, 43(3), pp.345-364.

Lundberg, S.M. and Lee, S.I., 2017. A unified approach to interpreting model predictions. In Advances in neural information processing systems (pp. 4765-4774).

Mao, C., Liu, Y. and Shen, Z.J.M., 2020. Dispatch of autonomous vehicles for taxi services: A deep reinforcement learning approach. Transportation Research Part C: Emerging Technologies, 115, p.102626.

McFadden, D., 1974. Conditional logit analysis of qualitative choice behavior. In: Frontiers in Econometrics, vols. 105&142. Academic Press, New York.

McFadden, D., 1978. Modeling the choice of residual location. Transp. Res. Rec. 673.v

McFadden, D., 1981. Econometric models of probabilistic choice. Structural analysis of discrete data with econometric applications, 198272.

Mckinley, J. and W. Hu. 2019. Congestion Pricing in Manhattan, First Such Plan in U.S., Is Close to Approval.

Miller, E. J., 2020. Travel demand models, the next generation: Boldly going where no-one has gone before. In Mapping the Travel Behavior Genome (pp. 29-46). Elsevier.

Miller, Eric J., 2018. Accessibility: measurement and application in transportation planning, Transport Reviews, 38:5, 551-555

Mokhtarian, P., 2018, August. Why travel surveys matter in the age of big data?. In 2018 National Household Travel Survey Workshop (p. 2).

Molnar, C., 2020, *Interpretable machine learning*. Lulu. com.

Newell, G.F., 1968. Queues with time-dependent arrival rates. III—A mild rush hour. Journal of Applied Probability, 5(3), pp.591-606.

Newell. G. F., 1982. Applications of Queueing Theory. Chapman and Hall/CRC, New York, second edition.

Nocedal, J., Wright, S., 2006. Numerical optimization. Springer Science & Business Media.

Nuzzolo, A. and Comi, A., 2016. Advanced public transport and intelligent transport systems: new modelling challenges. Transportmetrica A: Transport Science, 12(8), pp.674-699.

Olah, C., 2015. Calculus on computational graphs: Backpropagation.

Olah, C., 2015. Understanding lstm networks.

Ortuzar, J.D., Willumsen, L.G., 2001. Modelling Transport. Wiley, New York.

Paleti, R., Bhat, C.R. and Pendyala, R.M., 2013. Integrated model of residential location, work location, vehicle ownership, and commute tour characteristics. Transportation research record, 2382(1), pp.162-172.

Paliwal, M. and Kumar, U.A., 2009. Neural networks and statistical techniques: A review of applications. Expert systems with applications, 36(1), pp.2-17.

Paredes, M., Hemberg, E., O'Reilly, U. M., & Zegras, C., 2017, Machine learning or discrete choice models for car ownership demand estimation and prediction?. In 2017 5th IEEE International Conference on Models and Technologies for Intelligent Transportation Systems (MT-ITS) (pp. 780-785). IEEE.

Park, H., Haghani, A. and Zhang, X., 2016. Interpretation of Bayesian neural networks for predicting the duration of detected incidents. Journal of Intelligent Transportation Systems, 20(4), pp.385-400.

Paszke, A., Gross, S., Chintala, S., Chanan, G., Yang, E., DeVito, Z., Lin, Z., Desmaison, A., Antiga, L. and Lerer, A., 2017. Automatic differentiation in pytorch.

Pendyala, R.M., Konduri, K.C., Chiu, Y.C., Hickman, M., Noh, H., Waddell, P., Wang, L., You, D. and Gardner, B., 2012. Integrated land use–transport model system with dynamic time-dependent activity–travel microsimulation. Transportation Research Record, 2303(1), pp.19-27.

Pineda, P., 2019. Lime pulls electric scooters from Tempe, citing fees, liability burden

Rall, L. B., & LB, R., 1981. Automatic differentiation: Techniques and applications.

Raney, B., Cetin, N., Völlmy, A., Vrtic, M., Axhausen, K. and Nagel, K., 2003. An agent-based microsimulation model of Swiss travel: First results. Networks and Spatial Economics, 3(1), pp.23-41.

Recht, Ben., 2016. Mates of Costate. http://www.argmin.net/2016/05/18/mates-of-costate/

Refaeilzadeh, P., Tang, L. and Liu, H., 2009. Cross-Validation. Encyclopedia of database systems, 5.

Ribeiro, M.T., Singh, S. and Guestrin, C., 2016, August. " Why should i trust you?" Explaining the predictions of any classifier. In Proceedings of the 22nd ACM SIGKDD international conference on knowledge discovery and data mining (pp. 1135-1144).

Rumelhart, D.E., Hinton, G.E. and Williams, R.J., 1988. Learning representations by back-propagating errors. Cognitive modeling, 5(3), p.1.

Ryu, S., Chen, A. and Choi, K., 2017. Solving the combined modal split and traffic assignment problem with two types of transit impedance function. *European Journal of Operational Research*, *257*(3), pp.870-880.

Safikhani, A., Kamga, C., Mudigonda, S., Faghih, S.S. and Moghimi, B., 2018. Spatial-temporal modelling of yellow taxi demands in New York City using generalized STAR models. International Journal of Forecasting.

Saitz, J., 1999. Newton-Raphson method and fixed-point technique in finite element computation of magnetic field problems in media with hysteresis. IEEE Transactions on Magnetics, 35(3), pp.1398-1401.

Samek, W., Wiegand, T., & Müller, K. R., 2017. Explainable artificial intelligence: Understanding, visualizing and interpreting deep learning models. arXiv preprint arXiv:1708.08296.

Sanders, A. and Guse, C., 2019. NYC to impose some of the world's toughest regulations on Uber and Lyft.

Sarle, Warren S. 1994. Neural networks and statistical models.

Sbayti, H., Lu, C.C. and Mahmassani, H.S., 2007. Efficient implementation of method of successive averages in simulation-based dynamic traffic assignment models for large-scale network applications. Transportation Research Record, 2029(1), pp.22-30.

Schneider, T., 2015. Analyzing 1.1 billion NYC taxi and Uber trips, with a vengeance.

Shannon, C. E., 1948. A mathematical theory of communication. The Bell system technical journal, 27(3), 379-423.

Sheather, S., 2009. A modern approach to regression with R. Springer Science & Business Media.

Sheffi, Y., 1985. Urban transportation networks (Vol. 6). Prentice-Hall, Englewood Cliffs, NJ.

Shen, Z.J.M., Feng, B., Mao, C. and Ran, L., 2019. Optimization models for electric vehicle service operations: A literature review. Transportation Research Part B: Methodological.

Sifringer, B., Lurkin, V. and Alahi, A., 2018, May. Enhancing discrete choice models with neural networks. In Proceedings of the 18th Swiss Transport Research Conference (STRC), Monte Verità/Ascona, Switzerland (pp. 16-18).

Sifringer, B., Lurkin, V. and Alahi, A., 2020. Enhancing discrete choice models with representation learning. Transportation Research Part B: Methodological, 140, pp.236-261.

Smith, S., Fong, A., Yang, C. and Gardner, B., 2018. TravelWorks Integrated Models (No. DOT-VNTSC-FHWA-18-11). United States. Federal Highway Administration. Office of Planning.society for Industrial and Applied Mathematics, 8(4), 703-712. statistics models. International Journal of Education and Management Engineering, 6(3), pp.20-32.

Sun, J., Guo, J., Wu, X., Zhu, Q., Wu, D., Xian, K. and Zhou, X., 2019. Analyzing the Impact of Traffic Congestion Mitigation: From an Explainable Neural Network Learning Framework to Marginal Effect Analyses. Sensors, 19(10), p.2254.

Sun, L. and Axhausen, K.W., 2016. Understanding urban mobility patterns with a probabilistic tensor factorization framework. Transportation Research Part B: Methodological, 91, pp.511-524.

Sun, L., Erath, A. and Cai, M., 2018. A hierarchical mixture modeling framework for population synthesis. Transportation Research Part B: Methodological, 114, pp.199-212.

Train, K.E., 2009. Discrete choice methods with simulation. Cambridge university press.

van Kesteren, E.J. and Oberski, D.L., 2019. Structural equation models as computation graphs. arXiv preprint arXiv:1905.04492.

Verbas, İ.Ö., Mahmassani, H.S., Hyland, M.F. and Halat, H., 2016. Integrated mode choice and dynamic traveler assignment in multimodal transit networks: mathematical formulation, solution procedure, and large-scale application. Transportation Research Record, 2564(1), pp.78-88.

Vij, A. and Walker, J.L., 2016. How, when and why integrated choice and latent variable models are latently useful. Transportation Research Part B: Methodological, 90, pp.192-217.

Wang, F., Wang, J., Cao, J., Chen, C. and Ban, X.J., 2019. Extracting trips from multi-sourced data for mobility pattern analysis: An app-based data example. *Transportation Research Part C: Emerging Technologies*, *105*, pp.183-202.

Wang, R., Fan, S. and Work, D.B., 2016. Efficient multiple model particle filtering for joint traffic state estimation and incident detection. Transportation Research Part C: Emerging Technologies 71: 521-537.

Williams, H.C., 1977. On the formation of travel demand models and economic evaluation measures of user benefit. Environment and planning A, 9(3), pp.285-344.

Wong, M. and Farooq, B., 2019. ResLogit: A residual neural network logit model. arXiv preprint arXiv:1912.10058.

Wright, S. and Nocedal, J., 1999. Numerical optimization. Springer Science, 35(67-68), p.7.

Wu, X., Guo, J., Xian, K. and Zhou, X., 2018. Hierarchical travel demand estimation using multiple data sources: A forward and backward propagation algorithmic framework on a layered computational graph. Transportation Research Part C: Emerging Technologies, 96, pp.321-346.

Yang, C. and Gonzales, E.J., 2014. Modeling taxi trip demand by time of day in New York City. Transportation Research Record, 2429(1), pp.110-120.

Yao, J., Chen, A., Ryu, S. and Shi, F., 2014. A general unconstrained optimization formulation for the combined distribution and assignment problem. Transportation Research Part B: Methodological, 59, pp.137-160.

Ye, X., Konduri, K., Pendyala, R.M., Sana, B. and Waddell, P., 2009, January. A methodology to match distributions of both household and person attributes in the generation of synthetic populations. In 88th Annual Meeting of the Transportation Research Board, Washington, DC.

Zhang, Z., He, Q. and Zhu, S., 2017. Potentials of using social media to infer the longitudinal travel behavior: A sequential model-based clustering method. Transportation Research Part C: Emerging Technologies 85: 396-414.

Zhao, P., Liu, X., Kwan, M.P. and Shi, W., 2020. Unveiling cabdrivers' dining behavior patterns for site selection of 'taxi canteen'using taxi trajectory data. Transportmetrica A: Transport Science, 16(1), pp.137-160.

Zhao, X., Yan, X. and Van Hentenryck, P., 2019. Modeling Heterogeneity in Mode-Switching Behavior Under a Mobility-on-Demand Transit System: An Interpretable Machine Learning Approach. arXiv:1902.02904.

Zhou, X. and Taylor, J., 2014. DTALite: A queue-based mesoscopic traffic simulator for fast model evaluation and calibration.

Zhou, X., 2021. GitHub repository, https://github.com/asu-trans-ai-lab/asu-trans-ai-lab.github.io

Zhou, X., Mahmassani, H.S. and Zhang, K., 2008. Dynamic micro-assignment modeling approach for integrated multimodal urban corridor management. *Transportation Research Part C: Emerging Technologies*, *16*(2), pp.167-186.

Zhou, Z., Chen, A. and Wong, S.C., 2009. Alternative formulations of a combined trip generation, trip distribution, modal split, and trip assignment model. European Journal of Operational Research, 198(1), pp.129-138.

BIOGRAPHICAL SKETCH

Taehooie Kim is a research scientist at UrbanSim, focusing on the development of transportation planning models with Python data science and machine learning algorithms. He started pursuing the doctoral degree in Civil Engineering at Arizona State University under the guidance of Dr. Ram M. Pendyala and Dr. Xuesong (Simon) Zhou. He received the degree of Master of Science in Mechanical Engineering from Washington State University in 2017, and the degree of Bachelor of Science in Bio-Mechatronics in 2013 from Sung Kyun Kwan University, South Korea. His research interests lie in the integration of machine learning algorithms with conventional transportation planning models and the development of comprehensive modeling structures to map travel demand and network models. Publications of which Taehooie Kim is author include "A stepwise interpretable machine learning framework using linear regression (LR) and long short-term memory (LSTM): City-wide demand-side prediction of yellow taxi and for-hire vehicle (FHV) service", Transportation Research Part C: Emerging Technologies, 2020; and "Computational Graph-based Framework for Integrating Econometric Models and Machine Learning Algorithms in Emerging Data-Driven Analytical Environments", Transportmetrica A: Transport Science, 2021. He has designed transportation modeling software tools including, CGChoice – an open-source discrete choice model estimation, CGNetwork – an open-source origin-destination (OD) matrix estimation, and SIML – an open-source linear regression-based interpretable machine learning structure.