The Pennsylvania State University

The Graduate School

STATISTICAL LEARNING OF DYNAMIC TRANSPORTATION MODELS

A Dissertation in

Industrial Engineering and Operations Research

by

Wenjing Song

Submitted in Partial Fulfillment
of the Requirements
for the Degree of

Doctor of Philosophy

December 2020

The dissertation of Wenjing Song was reviewed and approved by the following:

Terry L. Friesz
Harold and Inge Marcus Chaired Professor
Dissertation Advisor
Chair of Committee

Enrique del Castillo
Distinguished Professor and Professor of Statistics

Uday V. Shanbhag
Gary and Sheila Bello Chair Professor

Vikash V. Gayah
Associate Professor

Runze Li
Eberly Family Chair Professor

Steven Landry
Professor
Department Head and Professor

ABSTRACT

Statistical learning (machine learning) and big data are among the most rapidly growing fields in the 21st century. Big-data based technologies emerge fast to provide business insight, push the boundaries of traditional communication, support informed decisions, and improve healthcare services. Statistical models are developed, revised, and applied every day to cope with the challenges big data brings.

In the big data era, well-developed statistical models are crucial to meeting the needs involved in studying transportation systems and the associated management challenges that accompany vehicular traffic operations. Modern traffic management emphasizes smart signal controls, automated driving, tolling, congestion relief, and emergency support during critical events of a public health and/or security nature. These traffic management tasks are intrinsically dynamic. In the field of transportation research, statistical learning methods are being increasingly applied to aid traffic management and forecasting. However, existing application of statistical learning to transportation modeling neither overcomes the need to mathematically articulate models nor resolves the curse of dimensionality that plagues all large-scale models. Rather, these studies provide alternative ways to make predictions based on already established models; those alternatives reduce the burden of finding new solutions when fundamental parameters change or data is insufficient. In this dissertation, we conduct original statistical learning studies on some of the most difficult traffic modeling problems with the aim of enhancing both computational efficiency and analytical simplicity. We present surrogate models, coupled to established traffic assignment and path finding procedures, that provide a

family of new statistics-assisted dynamic transportation modeling methods. The new framework that incorporates statistical models into DTA constitutes a paradigm shift. By taking a statistics-oriented direction, existing dynamic traffic assignment (DTA) models can be upgraded following our new framework without fundamental difficulty. This includes dynamic network loading (DNL), dynamic user equilibrium (DUE), and bi-level models for transportation network design and control.

DTA models rely on a network performance module known as dynamic network loading (DNL), which expresses flow propagation, flow conservation, and travel delay on a network level. DNL determines the so-called network delay operator, which maps a set of path departure rates to a set of path travel times or costs (delays). It is well known that the delay operator is not available in closed form and has undesirable properties that severely complicate DTA analysis and computation, such as discontinuity, non-differentiability, nonmonotonicity, and computational inefficiency. Given the theoretical and computational limitations of the conventional way of exploring the delay operator, we first propose a fresh take on this classic problem from the novel perspective of statistical metamodeling. Development of a DNL metamodel is the main focus of the first part of this dissertation, and the core technique on which the subsequent studies are built. In the DNL metamodeling part, our goal is to provide a class of surrogate DNL models that approximate the exact ones, with considerable benefits, including closed-form representation, improved regularity, and superior computational efficiency, at the expense of minor yet controllable approximation errors. Successful metamodeling of the DNL submodel opens a pathway to a family of new network performance models with a level of tractability not generally seen in conventional DNL; the result is a tool for improving

the analytical and computational experience associated with various classes of dynamic transportation problems. Any model that involves evaluating travel time on a network whose agents behave in an intrinsically dynamic way would benefit from the paradigm presented in this dissertation. Furthermore, we propose to apply our DNL metamodel to a group of classical dynamic transportation problems that require a delay operator, taking advantage of the closed form representation and analytical properties of the metamodel over conventional non-closed form DNL procedures. These applications include the reformulation of an approximate DUE with a closed-form delay operator and bi-level optimization problems with DUE embedded in its lower level.

We also introduce network aggregation through clustering and provide alternative covariance functions. These developments enable several extensions on the statistical metamodeling framework to help it accommodate a wider range of traffic models, including large-scale network models and bi-level DTA. Moreover, these extensions still enjoy all the desirable properties of the original model. We provide in-depth discussions on the implications of these properties in DTA research.

This dissertation contains eight chapters. In Chapter 1-3 we review dynamic transportation models and Kriging/metamodeling. In Chapter 4 we present the novel approach to build surrogate models for DNL using statistical metamodeling. Chapter 5 contains extensions of the notions of a distance metric, covariance function and modeling framework. Chapter 6 and Chapter 7 discusses applications to large-scale networks and bi-level problems. In the final chapter, we summarize the new statistical models presented for handling the challenges brought by big data in conjunction with DTA and suggest future research directions.

TABLE OF CONTENTS

LIST OF FIGURES ix

LIST OF TABLES x

ACKNOWLEDGMENTS xi

DEDICATION xv

Chapter 1 Dynamic transportation models 1

- Dynamic Traffic Assignment (DTA) 2
- Dynamic User Equilibrium (DUE) 10
- Dynamic System Optimal (DSO) 11
- Dynamic Network Loading (DNL) 12
- Motivation and goal of our proposed study 16
- Networks 17
- Notations 21

Chapter 2 Dynamic User Equilibrium (DUE) 24

- Mathematical formulation of Dynamic User Equilibrium (DUE) 27
- Continuous-time SRDT DUE 29
- DUE in discrete time 30
- Classifying DUE models 31
  - Route-choice (RC) only DUE 31
  - Simultaneous route-and-departure-time (SRDT) choice DUE 32
  - Bounded rationality 32
  - Elastic demand 33
  - Instantaneous/reactive DUE 34
- Computation of DUE 35
  - Fixed-point computational scheme 35
- DTA simulation 36

Chapter 3 Kriging/Metamodeling 39

- History of Kriging 39
- Categories of metamodeling 40
  - Gaussian process 40
  - Kriging 41
  - Spatio-temporal Kriging 44
  - Other metamodeling methods 45
- Metamodel Properties 46
  - Separability 46
  - Stationarity in space or time 47
  - Fully symmetry 47
- Metamodeling on Dynamic Network Loading (DNL) 48

Chapter 4 Statistical metamodeling on DNL 49

Metamodeling in DNL 51
Dynamic network loading and the delay operator 51
Delay operator as an infinite-dimensional mapping 52
Delay operator as a finite dimensional mapping 53
Methodology-metamodel and experimental design 55
Training and predicting procedures 58
Experimental design of training dataset 59
Improvement in experimental design and sampling 63
Numerical studies 64
Test scenarios 64
Test results for the Kriging-based delay operator 66
Comparison to Neural Networks 70
Mathematical properties 71
Closed-form expression of the delay operator $\mathbf{\Phi}$ 72
Analytical properties of the surrogate delay operator $\mathbf{\Phi}$ 73

Chapter 5 Extensions: Distance metrics, covariances, and spatio-temporal Kriging 76

Distance metrics 76
Regular similarity for Kriging 78
Revised similarity for Kriging 79
SLS similarity with higher degree 80
Affine similarity metric 80
Benefit of new distance metrics 81
Validity 82
Covariance functions 83
Spatio-temporal Kriging 85

Chapter 6 Large-scale transportation networks 87

Dimension reduction and improvement in model training speed 87
Clustering 89
Framework of network aggregation 90
Path aggregation 90
Creation of path groups 91
Aggregated path 91
Framework and formulation of aggregation 94
Algorithms of aggregation 96
Algorithm 1 A heuristic of path aggregation based on SLS 96
Algorithm 2 Grouping into predefined number of groups m 97
Algorithm based on K-means 97
Network aggregation by link 98
Numerical study framework 99
Picture of the network 99
Chicago sketch 100

Chapter 7 Bi-level problems and MPEC 102

Introduction of bi-level problems....102
A hybrid model for dynamic user equilibrium (DUE) computation....102
Fixed-point method with DNL metamodel....103
Application of DNL metamodel on bi-level traffic models....105
A bi-level signal control problem....106

Chapter 8 Discussion....109

Significance....112
Summary....114

Appendix A Mathematical proof on validity of distance metric....117

Appendix B Mathematical proof on validity of covariance function....121

Reference....124

LIST OF FIGURES

Figure **1-1**: Network of Nguyen. 18

Figure **1-2**: Network of Sioux Falls. 19

Figure **1-3**: Network of Chicago Sketch. 20

Figure **3-1**: Illustration of Kriging 40

Figure **3-2**: Illustration of Kriging on an X→Y mapping 41

Figure **3-3**: Metamodeling on delay operator. 47

Figure **4-1**: Illustration of the DNL model 52

Figure **4-2**: Illustration of the mapping from h_p to Ψ_p when $n = 8$, which defines the finite dimensional operator Ψ 54

Figure **4-3**: Prediction trend of Nguyen network: prediction errors and training time as size of training dataset increases 71

Figure **4-4**: The Nguyen network: comparison of exact and predicted path delays on example paths. 67

Figure **4-5**: Prediction trend of Sioux Falls network: prediction errors and training time as size of training dataset increases 68

Figure **4-6**: Sioux Falls network with 501 paths: comparison of exact and predicted path delays on example paths 69

Figure **4-7**: Comparison of Kriging and NN on the Nguyen network: the vertical bars denote two standard deviations from the prediction means. 70

Figure **6-1**: Path aggregation illustrative example: network 91

Figure **6-2**: Single Highway network 98

LIST OF TABLES

Table **1-1**: Statistical learning applied to transportation models in literature 17

Table **4-1**: Performances of the surrogate delay operator Φ: the prediction errors and speed-up indices are averaged over 100 randomly sampled testing data. 66

Table **5-1**: Studies that discuss or apply non-Euclidean distance metric 76

Table **6-1**: Path aggregation illustrative example: path information 91

Table **6-2**: Single Highway network data 98

Table **6-3**: Chicago Sketch network data 99

ACKNOWLEDGMENTS

I would like to start with a distinguished thank you to my advisor Dr. Friesz and my committee members Dr. del Castillo, Dr. Shanbhag, Dr. Gayah and Dr. Runze Li. My primary advisor Dr. Friesz has been long-term mentor and major supervisor of my research and a role model of the passion a scientist has for his/her career. He is remarkably trustworthy and is respectful when interacting with students. Dr. del Castillo provided substantial guidance from his expertise in statistics. He is a great teacher in class and has greatly inspired me with his passion for training young minds. I am deeply impressed by his devotion in conveying statistical ideas which makes his class so much fun. Both Dr. Friesz and Dr. del Castillo are researchers who are highly skillful in critical thinking and emphasize the quality of their work. Their high standard for both themselves and students make me constantly feel the pressure to improve. They enjoy my special thanks.

My sincere gratitude must be presented here to Dr. Ke Han who is not officially on my committee, but mentoring me shoulder to shoulder throughout my doctorate voyage. His non-stop strong support and recognition is outstanding and his devotion to academic is dramatically encouraging. My gratefulness for him is endless. His expertise in DNL and DTA models lead me to a comprehensive and clear understanding of the problem I am proposing to solve in my study, and teach me to look at hard problems strategically and emphasize the importance of perspective. Discussing with him provided me with immediate new inspirations in looking at the bigger picture and discovering new pathways to make the scientific approach stronger. Working with him has helped me build confidence in my ability to challenge difficult questions and pursue goals with confidence and care. He psychologically inspired me to trust in myself and break limitation when I am down, and remind me of my own strength and value. His support

encouraged me to steadily go on the path toward my dream career in the academia and achieve more professional and personal goals in my list of dreams.

Dr. Runze Li always provides substantial help and new perspectives. He is a remarkably knowledgeable and very humble professor with whom I always enjoyed and benefit talking to. He builds a figure of great researcher for me who loves his work deeply, care for students from his heart and willing to help students.

Dr. Shanbhag and Dr. Gayah joined my committee with support and encouragement at the critical time point of my graduation. Dr. Shanbhag is a very humorous professor and always patient in helping students. His expertise in optimization is iconic and I am honored to have him on my committee. Through the short period I work with Dr. Gayah, I can feel his passion and positive thinking. I feel seen and welcome communicating with him, which is very encouraging. I am grateful for both of their kind help and advice.

I am grateful for the financial and academic support Dr. Tao Yao provided. Dr. Yao has been my co-advisor during the most of my doctorate study. His support along with Dr. Friesz enable me to initiate my doctorate research at Penn State and grow more skills along the journey of pursuing PhD. Dr. Yao left Penn State before I graduate, therefore is not listed as a committee member. This does not make him a less important professor than the listed committee members in my education and growth. He also enjoys my special thanks.

I want to thank my friends and fellow graduate students for their emotional support provided at my dearest need, working with me through hard time both academically and personally. They are Xiao Wei (Dr.- to be), Hongyi Sun, Qing Liu, Fang (Xingwei) Qi, Wen Shan, Jing Wang (Dr. -to be), Sihong Yan (Dr. -to be), Leila Ziqi Wu (Dr. -to be), Dr. Lulu Bao, Dr. Ying Dong, Dr. Xiangyu Li, Dr. Haoyun Zhang, Xiaochuan Ma (Dr. -to be), Dr. Xue Wang, Dr. Yue Xie, Dr. Hongcheng Liu, Dr. Hongsheng Zhang, Dr. Gang Li, Xiaofei Chen, for their suggestions and support. I want to thank Lixuan Zhao, Dr. Hongcheng Liu, Dr. Yiou Wang and

Dr. Ke Han for proofreading this dissertation. I thank Dr. Xue Wang for his remarkable supportiveness and encouragement. I thank my furry friends Huamao Song and Huaniu for their adorable emotional support provided through their lovable fur and happy jumps.

I thank my parents and cousin for the strong family support they provide whenever I am in need. My mother, a very opened minded women, a retired engineer and remarkably caring family member is the primary daily supporter and first audience of many of my research approaches. Her wisdom of making life choices are always inspiring. My family are happy to sacrifice their own priority and resting time to help me enthusiastically which makes me both encouraged and humble to learn more and always acknowledge that there are a lot more human do not know than we do, and the fantastic mystery of nature is always there for generations of researchers to work feverishly on.

I appreciate my medical doctors highly for medical support and healthcare advice they provide and several anonymous doctors for confidential reasons at Peking Union Medical College Hospital, Penn State Health Center and Healthiest You healthcare network. They make it possible to help me come back from pain, re-build my health, and improve my understanding of being a human and myself. I now view research as work of intelligence and achievement as well as humanity and part of my inner strength.

I thank , Danielle Fritchman especially for her emails are always warm and friendly. I am also thankful to the PEO organization for financial support in 2016 and 2017, and warm encouragement throughout my study. I thank Penn State University and the local community for providing me with unforgettable studying and living experience.

One person I must not forget to acknowledge is myself, for my persistence, passion, and self-support in achieving this milestone. From my personal experience, the popular Chinese concept that “the biggest enemy is yourself” is very insightful, but I discovered that the concept that “the biggest friend is yourself” is more important. This achievement improved my

knowledge, skill, and intelligence; ability to work hard, mental power, and communication. I am proud of myself for not giving up my dream facing stress, difficulties, sickness, and life incidents. I am strong.

I thank Dr. Friesz and department of IE for providing financial support from 2014 to 2018 (NSF funding award number 1300638 is included in partial funding of the author). I express my thanks to the National Science Foundation. I confirm that, as required by federal guidelines, all federal funds used for the work described in this dissertation have been explicitly acknowledged in the document, along with a disclaimer indicating that the findings and conclusions do not necessarily reflect the view of the funding agency.

DEDICATION

This dedication of this dissertation is to all girls and women who have a dream to become a doctor and a scientist, if you have stick with your dream until you realize it.

Chapter 1

Dynamic transportation models

In this chapter we discuss models of Dynamic Traffic Assignment (DTA), Dynamic User Equilibrium (DUE), and Dynamic Network Loading (DNL) in transportation modeling.[1]

Modern transportation models largely take the dynamic view compared to a static one, considering that static models are criticized for its over simplification of the road network traffic system which is, in nature, dynamic in time. Therefore, we will only work on the dynamic models in this proposed dissertation study, with necessary review on basic concepts that originate from their static counterpart.

In the related literature, the frequently discussed dynamic transportation models include but are not limited to dynamic traffic assignment, dynamic freight transportation problems, dynamic network design. Some models combine multi-layers of dynamic transportation problems. In this dissertation, we focus on the DTA and dynamic network design problem (bi-level problems). Among dynamic transportation models, these are the most frequently visited and mostly emphasized.

A dynamic traffic assignment describes traffic flow on a network that changes over time. The DTA models are of primary focus of this dissertation. Specifically, when we look into the individuals of travelers on a traffic network they can be viewed as independent players in a Nash Game. Nash Game is a concept named after the mathematician John Forbes Nash Jr. A Nash Game is defined as a non-cooperative game where each user has independent set of strategy, and

[1] Some of the content presented in this chapter were previously presented in Song et al. (2018).

the payoff function of each user is depending on everyone's choices. We will in detail introduce the DUE model which base on this Nash-like Game view of traffic assignment.

In literature, developed DTA models follows inspiration from Wardrop through his paper (Wardrop, 1952) in 1952 on dynamic extensions of User Equilibrium (UE) and System Optimal (SO). In 1993, Friesz et al. (1993) introduced the variational inequality representation of the DUE problem, which brings revolution in the computation and analysis of DUE.

Dynamic Traffic Assignment (DTA)

This section introduces the history of static and DTA models. Traffic assignment is the behavior-based allocation of origin and destination-specific, forecasted travel demands to the arcs of a real physical road network (Friesz and Bernstein, 2015). Intrinsically, traffic assignment modeling is a descriptive modeling. It is fundamentally a problem of route choice. DTA is a family of assignment models that includes notions of DUE and dynamic system optimal. To date, there have been many papers and a few books that contain summaries of DTA research published. To provide the background context for this dissertation, we present the following significant review articles for DTA and specific areas of DTA topics. These articles also present the intellectual history of DNL, dynamic system optimal (DSO), and DUE. They are (Han et al., 2019; Friesz et al., 2018) the following:

1. Cascetta and Cantarella (1993). Puts forward a general framework for dynamic simulation in transportation networks. It presents the models and algorithms for both within day and day-to-day DTA. The framework it puts forward covers static models as a particular case, and it is universal enough so as to encompass a large number of existing DTA models.

2. Ran and Boyce (1996a): Contains a discussion of DTA from the perspective of intelligent transportation system.
3. Peeta and Ziliaskopoulos (2001). Takes large numbers of DTA formulation and solutions into consideration, including those on the basis of the forms of mathematical programming, variational inequalities, optimal control, and simulation. Their papers record the main DTA approaches by publication for future reference, generalize the understanding of DTA at that time, review the previous literature, and formulate a hypothesis about the future of the discipline.
4. Boyce et al. (2001). Contains discussion on analytical models of the DTA problem. Takes a limited subset of so-called analytical formulations of the DTA problem, with a focus on the authors' experience with variational inequalities. It identifies the solution algorithms and computational issues which need to be further studied.
5. Szeto and Lo (2005a). A DTA review that emphasizes the fidelity and accuracy of DTA models for online network planning, policy evaluations, and real-time operation and management. Summaries and analyzes DTA formulations. It also makes a comparison with the application of DTA formulations concerning their use to traffic flow theory and suggests future research directions.
6. Szeto and Lo (2005b). Focuses on properties of DTA with physical queues. Only focuses on the literature which deals with physical queues, and recommends spillback models rather than point-queue concepts for DTA. Compares the characteristics of physical-queue DTA with point-queue DTA, stressing the first-in-first-out (FIFO) [2]queue discipline and queue spillback, as well as the causal relationship and travel-time-link-w consistency.

[2] FIFO: first-in-first-out rule, which assume that travelers enter the network early also exit early.

7. Mun (2007). Contains a review of traffic performance models for DTA, and analyzes the advantages and disadvantages of existing models. It identifies the requirements for traffic performance models and discusses different forms of existing traffic performance models for DTA. The weaknesses of nonlinear travel-time models may make them inappropriate for the analysis of time-varying transportation networks, and the linear type travel-time models also show some limitations. Therefore, it can be seen that there exists a dilemma, and the settlement of it is key to the theoretical coherence and credibility of DTA modeling.
8. Rakha and Tawk (2009). Traffic networks: dynamic traffic routing, assignment, and assessment. Presents dynamic travel behaviors and relevant models. It is argued that such behavioral models are vital to DTA Modeling. It also speculates what kind of models will be needed in the future and their fundamental properties.
9. Viti and Tampere (2010). Discusses real time applications and realistic user behavior on networks. Conjectures future directions of model building and their essential properties needed. This paper contains a selected collection of refereed papers from the DTA modeling.
10. Jeihani (2010). In this paper the authors review the DTA models included in some famous computer packages. It presents demand estimation, supply presentation, methods for computing dynamic user equilibria, and convergence mainly focusing on TRANSIMS, which reckons second by-second movements of individual travelers while employing parallel processing and cellular automata representations. It finds that TRANSIMS can deal with some existing problems in dynamic traffic assignment models, but it cannot overcome the problems of existence, stability, uniqueness and other important qualitative properties.
11. Chiu et al. (2011): Presents a primer on simulation-based DTA modeling.

12. Szeto and Wong (2012). Dynamic traffic assignment: model classifications and recent advances in travel choice principles. Pays attention to the principle of travel choice and the categorization of DTA models. Discusses the connotations of the so-called travel choice principle for existence and uniqueness of DTA solutions. The nonlinear complementarity, variational inequality, mathematical programming, and fixed-point model formulations are used to expound the relationship between the travel choice principle and traffic flow exploiting.
13. Garavello et al. (2016): Reviews the traffic flow modeling aspect of DTA, namely the hydrodynamic models for vehicular traffic and their network extensions.
14. Wang et al. (2018): Reviews relevant DTA literature concerning environmental sustainability.

Wardrop's first and second principle

In 1952, the British transport analyst Wardrop proposed two principles that are later referred to as Wardrop's first and second principle (Wardrop, 1952). These two principles provide the foundation of the static traffic assignment theories for road networks. In Wardrop's First Principle, also called user optimal principle, travelers are assumed to act like Nash players who want to minimize their own disutility (the travel cost) non-cooperatively by adjusting their route choice, and the traffic system reaches UE when no traveler can achieve a lower perceived travel cost by unilaterally changing his/her route. Wardrop's Second Principle, also named system optimal principle, assumes traveler to behave cooperatively, and the traffic systems is said to reach SO when the total travel time of all travelers is minimized. UE is believed to be a more appropriate modeling framework than SO for road passenger network without central control, and the SO principle is most frequently addressed in problems that involve centralized control, such

as freight transport problems while being relatively less relevant to road passenger network. Proposed study in this dissertation primarily concerns UE and its extensions.

Static assignment models and dynamic extensions

From 1950s to 1980s, much research effort has been put to the development of static network equilibrium transportation models, analysis and algorithms (Dafermos, 1980, 1982a, 1982b, 1983; Evans, 1976; Friesz, 1983). A static user equilibrium is defined as follows.

Definition 1.1. (Static user equilibrium) a flow pattern (f, u) is a static user equilibrium when it satisfies

$$h_p\left[C_p(h) - u_{ij}\right] = 0 \ \ \forall\left(i, j, p \in \mathcal{P}_{ij}\right),$$

$$C_p(h) - u_{ij} \geq 0 \ \ \forall\left(i, j, p \in \mathcal{P}_{ij}\right),$$

$$\sum_{p \in \mathcal{P}_{ij}} h_p - T_{ij}(u) = 0 \ \forall(i, j),$$

$$f_a - \sum_{\mathcal{P}} \delta_{ap} h_p = 0 \ \forall a,$$

$$h \geq 0,$$

$$u \geq 0.$$

Regard the conditions listed above, T_{ij} denotes the demand for transportation between origin i and destination j; f_a denotes traffic flow on arc a; $f = (\ldots f_a, \ldots)$ denotes the vector of all arc flows; h_p is the flow on path p; δ_{ap} is defined to be 1 if arc a belongs to path p and 0 otherwise; and the average cost of transportation on path p is denoted by $C_p(h)$.

In the definition of static user equilibrium, the first two equations in the conditions for (f, u) are equivalent to Wardrop's First Principle (Wardrop, 1952). Dafermos (1982a) discussed static version of assignment and showed that the static user equilibrium conditions are completely

equivalent to a Variational Inequality (VI) problem, which provide the foundation of using the VI to formulate traffic assignments in dynamic form. The VI formulation of a static UE is as follows (Dafermos 1982a; Friesz, 1983)

$$find\ (f^*, T^*) \in \Omega, such\ that$$

$$c(f^*)(f - f^*) - \theta(T^*)(T - T^*) \geq 0$$

$$\forall (F, T) \in \Omega$$

where

$$\theta(T) = (\dots, \theta_{ij}(T), \dots)$$

$\theta_{ij}(T)\ is\ the\ inverse\ transportation\ demand\ function\ for\ OD\ pair\ (i,j)$ and

$$\Omega = \left\{ (f,T) \colon \sum_{p \in \mathcal{P}_{ij}} h_p - T_{ij} = 0\ \ \forall (i,j); f_a - \sum_{\mathcal{P}} \delta_{ap} h_p = 0\ \ \ \forall a;\ h \geq 0; T \geq 0 \right\}.$$

Dafermos's work shows that the static user equilibrium can be formulated as an VI. This important development directly inspired researchers to a new direction that powerful theorems for VIs may be employed for solving the assignment problems and to establish the qualitative properties of a user equilibrium. Friesz (1983) provides a succinct review of key developments of traffic assignment and equilibrium in the aforementioned three decades. The review visited static transportation network equilibrium modeling and the related fields of network design and network aggregation. Friesz (1983) points out that the static (or steady state) network equilibrium studied and applied in transportation practices were unrealistic in several respect, and the foremost one is the absence of dynamic considerations. Several improvements in fundamental levels and model construction were identified to enhance the predictive capability of a transportation assignment model. Nevertheless, static user equilibrium provides solid foundation and largely influences the later developments of the dynamic models which is widely accepted today.

Merchant and Nemhauser (1978a) and Merchant and Nemhauser (1978b) was among the earliest attempt to extend static SO problem to dynamic version and formulate it as a mathematical program. They propose a discrete-time link performance model based on a link exit function (Han, 2013). The model is named after the authors as the M-N model. The M-N model assumes a functional relationship between the exit flow on a link and the number of vehicles present on the link (i.e. the link occupancy). The M-N model also uses a static link performance function to represent the travel cost as a function of link occupancy (Han, 2013). The publication of M-N model inspired and is followed by several in-depth studies of DTA problems in Carey (1986, 1987); Friesz et al. (1989) and Wie et al. (1995) (Han, 2013).

Dynamic traffic assignment (DTA)

DTA is the descriptive modeling of time-varying flows on traffic networks consistent with traffic flow theory and travel choice principles. DTA models describe and predict departure rates, departure times, and route choices of travelers over a given time horizon (Song et al., 2018). Analytical DTA models consist of two main components: (i) the mathematical expression of trip assignment such as the dynamic extension of the Wardrop's principles (Wardrop, 1952); (ii) the network performance model, which captures the relationships among link entry flow, link exit flow, junction flow, link delay, and path delay. The latter is usually referred to as DNL. The DNL problem gives rise to the delay operator, which is interpreted as a mapping from the set of path departure rates to the set of path travel times. Such delay operators will be one of the main focus of this dissertation, although other notions of the delay operators, often going by different names, have been invoked in a variety of different contexts (Gentile et al., 2007; Jang et al., 2005; Lo and Szeto, 2002b; Perakis and Roels, 2006; Ukkusuri et al., 2012).

In this field, much intellectual energy has been devoted to the DUE problem, which is a Nash-like non-cooperative differential game wherein agents minimize their effective travel delay through route selection and departure time choice for any given trip purpose. Some literature also conceptualizes and model DSO traffic assignment for traffic networks, where a centralized agency optimizes total system cost/time by all travelers.

The DNL subproblem is interpreted as a mapping from the set of path departure rates to the set of path travel times. Being an integral part of a complete mathematical formulation of DTA problems, the delay operator plays a fundamental role and affects the analytical properties of the DTA models in many different ways. Therefore, a great amount of literature has been published on modeling and computational methods for the DNL subproblem itself. A detailed introduction on DNL is given in the Dynamic Network Loading (DNL) section in this Chapter.

In the following 3 sections, we introduce the basic concepts and briefly review relevant literature of DUE, DSO, and DNL, respectively. In next chapter, we will introduce and review the DUE model in more details.

The DUE model assumes each user simultaneously makes two decisions: route choice and departure time choice. The origin and destination are determined and pre-assumed by the set of OD pairs. For models with elastic demand, total demand on a network is a variable. For other instances, the demand is assumed to be a constant. When each user chooses both departure time and route to take, the decision is a very flexible one. Overall traffic status by this setting is expected to have more complexity and of high non-linearity. Gridlocks, traffic jams, and spillback are expected phenomenon when flow grows high. Mathematically, this means increased complexity and challenges in tractability.

Dynamic User Equilibrium (DUE)

Dynamic user equilibrium (DUE) on a traffic network describes that route flow reaches equilibrium while each traveler makes their independent travel decision to optimize travel cost. The DUE model employs the scheme of non-cooperative game theory. DUE problems have been studied within the broader context of DTA, which is viewed as the modeling of time-varying flows on traffic networks consistent with established travel demand and traffic flow theory (Han et al., 2019). DTA models, from the early 1990s onward, have been greatly influenced by Wardrop's principles (Wardrop, 1952, Han et al., 2019).

The generalization from the static user equilibrium to the dynamic setting has been difficult in many aspects. Dynamic user equilibrium and its network performance aspect needs to be considered mathematically in infinite dimensional settings. Especially, network performance model in dynamic settings need to capture dynamic traffic flow phenomenon such as shockwaves and it was adding to the complexity of developing a convincing assignment model. Only with a realistic performance module that captures dynamic flows can the model properly predict traffic flows on the network in a dynamic way. Difficulties in the formulation is foremost, and these non-trivial aspects of the model follow immediately of the idea to take dynamic considerations in UE and SO. These extensions attracted many researchers to work on and the development was not immediate.

For user equilibrium traffic assignment, multiple formulations were developed based on a spam of mathematical methods: nonlinear complementarity problem, optimal control theory, etc. The successful generation of static to dynamic UE was achieved in the work by Friesz et al. (1993) by using VI formulation. The generation included route and departure time choice for each user on the network, and formulate the entire model in dynamic settings.

To date, there are multiple means formulating dynamic user equilibrium. Some of the formulation methods are (Han, 2013)

- a variational inequality (Friesz et al., 1993; Smith and Wisten, 1994, 1995)
- an evolution equation in an appropriate function space (Mounce, 2006; Smith and Wisten, 1995)
- a nonlinear complementarity problem (Wie et al., 2002; Han et al., 2011)
- a differential variational inequality (DVI) (Friesz et al., 2001, 2011, 2013b; Friesz and Mookherjee, 2006); and
- a differential complementarity system (Pang et al., 2011).

Dynamic System Optimal (DSO)

Another important class of dynamic transportation model is the DSO assignment model. DSO is viewed as an extension of Wardrop's Second Principle, also familiar from traditional, static traffic assignment for traffic networks. DSO seeks system-wide minimization of travel costs incurred by travelers, subject to the constraints of travel demand, link dynamics, flow propagation and travel delay (Han, 2013).

DSO problems enjoy the benefit of having a well-defined objective function. It can be minimizing total system cost or time. Hence, it is usually more amenable to analysis (Nie, 2011). Merchant and Nemhauser (1978a, b) initiated their DSO studies by presenting the mathematical formulation of the DSO problem in discrete time. This model is known as the M-N model. In later years, DSO studies built on and extend the work done by Merchant and Nemhauser. Birge and Ho (1993) extend the M-N model to a stochastic case, Ziliaskopoulos (2000) introduce a linear programming formulation for single destination DSO problem. Friesz et al. (1989) reformulated the M-N model as a continuous optimal control problem and interpreted the first-

order condition using a so-called instantaneous path marginal cost. Entering 1990s, Wie et al., (1994) proposed an iterative solution method for the continuous model. Nie (2011) summarizes more recent development in the DSO studies.

Solution of DSO problems has been used as a benchmark to support evaluating the benefits of several categories of policies and strategies. Some of them are the following:

- Investment decisions: network expansion (Waller, 2000; Karoonsoontawong and Waller, 2010)
- Traffic management policies: congestion pricing and information provision (May and Milne, 2000; Shen and Zhang, 2009)
- Operational strategies: signal control and ramp metering (Muñoz and Laval, 2006)
- Evacuation planning: large-scale evacuation planning (Sattayhatewa and Ran (2000); Jha et al., (2004); Han et al., (2005); Chiu et al., (2005); Sbayti and Mahmassani (2006); Liu et al., (2006); Shen et al., (2007))

Dynamic Network Loading (DNL)

In DTA modeling, the DNL subproblem is identifying how to describe and predict the spatio-temporal evolution of traffic flows and congestion on a network that is consistent with established route-and-departure-time choices of drivers. This is done by introducing appropriate dynamics to flow propagation, flow conservation, and travel delays on a network level. Any DNL must be consistent with the established path departure rates and link delay model, and is usually performed under the FIFO rule. As aforementioned, the DNL subproblem defines the network delay operator. A delay operator maps a set of path departure rates to a set of path travel times. Rigorous definitions of delay operator are given in Chapter 4. Existing DNL models developed in literature include:

- the link delay model (Friesz et al., 1993);
- the Vickrey model (Vickrey, 1969);
- the Lighthill–Whitham–Richards (LWR) model (Lighthill and Whitham, 1955; Richards, 1956);
- the cell transmission model (Daganzo, 1994, 1995);
- the link transmission (double-queue) model (Yperman et al., 2005; Han et al., 2015c); and
- point-queue model (Han et al., 2013a, b)[3].

Although there have been many different models for DNL subproblem, LWR is believed to be the only proper basis for DNL modeling because it properly captures shockwave propagation (Han, 2013). Because of its attractive feature, the LWR model has enjoyed lots of scholarly attention on its formulation, solution and extension for supporting dynamic assignment models (Blimer, 2007; Raadsen et al. 2016; Lo and Szeto, 2002a). The LWR model describes the spatial-temporal evolution of vehicle density on each link of the network using a scalar conservation law. It is well-established that a full LWR-based network model requires boundary conditions on both ends of the link, which account for the propagation of upstream/downstream information (Han, 2013). It follows from this structure that we require one LWR model for each link of the network along with boundary conditions that describe what must be done when flows of vehicle meet. Thus, we have a system of simultaneous partial differential equations and boundary conditions to solve. Such DNL models are intrinsically difficult to solve, making statistical learning an attractive alternative.

[3] In some transportation articles the term "point-queue" refers to completely different algorithms. Here we are referring to the point-queue DNL model in the cited literature specifically.

Being an integral part of a complete mathematical formulation of DTA problems, the delay operator plays a fundamental role and affects the analytical properties of the DTA models in many different ways. For instance, the existence of DUE[4], which is the most widely studied form of DTA problems, depends on the continuity of the delay operators (Han et al., 2013c; Smith and Wisten, 1995; Zhu and Marcotte, 2000). The uniqueness of DUE[5] is guaranteed by the monotonicity of the delay operator (Mounce and Smith, 2007). Looking at the computational point of view, all computational procedures for DUE problems rely on certain versions of continuity and monotonicity to converge (Friesz et al., 2011; Han et al., 2015; Long et al., 2013; Mounce, 2006; Szeto and Lo, 2004). Moreover, differentiability is typically required for the delay operator for problems such as sensitivity analysis (Chung et al., 2014; Yang and Bell, 2007) and mathematical programs with equilibrium constraints (Yang et al., 2007).

Studies of the DNL models date back to the 1990s with a significant number of publications (Friesz et al., 2013a; Lo and Szeto, 2002b; Szeto, 2003; Szeto and Lo, 2004; Ukkusuri et al., 2012). Among these advancements some have focused on capturing realistic network congestion effects such as the formation, propagation and dissipation of queues and spillbacks (Nie and Zhang, 2010; Han et al., 2015b) while others have managed to reduce the complexity of the network dynamics and improve the computational efficiency (Yperman et al., 2005; Gentile et al., 2007).

- *Limitations of traditional DNL problems*

For large-scale networks, and sophisticated traffic models that capture phenomena such as shock waves and car spillback, it has been recognized in studies that the DNL models are

[4] The existence and uniqueness of DUE refers to the existence and uniqueness of the equilibrium, respectively.

[5] See footnote above.

rather complicated, and the corresponding delay operators enjoy very few analytical properties essential for the analysis and computation of DTA models. For example, DNL with physical-queue models such as the Lighthill–Whitham–Richards model, cell transmission model, and link transmission model is found to yield discontinuities in the delay operator for general networks (Szeto, 2003; Han et al., 2015b). As a result, the delay operator is non-differentiable.

The nonmonotonicity of the delay operators on networks has been reported in the literature (Mounce and Smith, 2007) and is the major hurdle to computing DUE solutions with convergence guarantee. Due to aforementioned nature, it is not surprising to conclude that the delay operator is generally not available in closed form. Numerical evaluation via the DNL procedure is the most commonly adopted. Such a procedure is based on a series of link, node, and path dynamic models that typically involve solving ordinary or partial differential equations (Friesz et al., 2013a; Gentile et al., 2007; Lo and Szeto, 2002b; Perakis and Roels, 2006; Szeto and Lo, 2004; Yperman et al., 2005).

Expressing the complete DNL model analytically and embedding it into certain math programming formulation is therefore an onerous task, and could severely complicate the DTA computational procedures (Friesz et al., 2013a; Szeto and Lo, 2004; Ukkusuri et al., 2012). Moreover, conventional DNL procedures tend to be computationally demanding, and can scale badly as the network size increases. It is expected to be computational tractable for models to be practical. A computational tractable model can be solved by off-shelf commercial solvers easily. These limitations of traditional DNL models is identified and motivates our alternative approach by statistical metamodeling.

Motivation and goal of our proposed study

Given the aforementioned theoretical and computational limitations of the conventional way of exploring the delay operator, we first propose a fresh take on this classic problem from the novel perspective of statistical metamodeling. Development of a DNL metamodel is the main focus of the first part of this dissertation, and the core technique on which the subsequent studies are built. In the DNL metamodeling part, our goal is to provide a class of surrogate DNL models that approximate the exact ones, with considerable benefits including closed-form representation, improved regularity, and superior computational efficiency, at the expense of minor yet controllable approximation errors. Successful metamodeling of the DNL submodel will open a pathway to a family of new network performance models with tractability barely seen in existing ones, and provide a means of improving the analytical and computational method for various classes of dynamic transportation problems. Any model that involves evaluating travel time on a network of dynamic settings would benefit from doing so. We propose to apply the developed DNL metamodel to a group of classical dynamic transportation problems where delay operator is required, taking advantage of the close form representation and analytical properties of the metamodel over conventional non-close form DNL procedures. These applications include the reformulation of an approximate DUE with a closed-form delay operator and bi-level optimization problems with embedded DUE as lower levels.

To better explain our approach, we start from a review on DUE, Kriging, and statistical learning in the next section, followed by introduction of the specific metamodeling method we choose to use.

Literature	Metamodel	Application	Underlying model	Object of surrogate model
Ciuffo et al. (2013)	Ordinary Kriging	Sensitivity Analysis	Mesoscopic simulation (AIMSUN)	Network-wide density, average flow, average delay, average travel time

Zhang et al. (2014)	Stochastic Kriging	Active traffic management	Simulation-based DTA(DynasT)	Network average trip time
Chen et al. (2015)	Universal Kriging	Bi-level network design	Simulation-based DUE (DTAlite)	Network average travel time
Chen et al. (2014)	Polynomial, Gaussian RBF, Kriging	Bi-level road pricing	Static MPEC, AIMSUN simulation	Network average travel time
Idé and Kato (2009)	Ordinary Kriging	Travel-time estimation	Agent-based simulation (ABM)	Path travel time
Xie et al. (2010)	Universal Kriging	Short-term traffic flow forecast	Empirical data	Link traffic volume
Sun and Xu (2011)	Mixtures of Gaussian processes	Short-term traffic flow prediction	Empirical data	Link traffic volume
Chan et al. (2012)	Neural networks	Short-term traffic speed forecast	Empirical data	Link traffic volume
Wang and Shi (2013)	Hybrid Support Vector Machine	Short-term traffic speed forecast	Empirical data	Link traffic speed
Ye et al. 2012	Hybrid ARIMA and NN	Short-term traffic speed forecast	Empirical data	Road segment traffic speed

Table **1-1** Statistical learning applied to transportation models in literature

With regard to the table above these are the abbreviations that we use: RBF: radial basis function; ARIMA: autoregressive integrated moving Average; NN: neural networks; MPEC: mathematical programming with equilibrium constraints; ABM: agent-based model. AIMSUN, DynasT and DTAlite are names of the software used in conducting dynamic traffic assignment and simulation.

Networks

In our dissertation, we provide numerical examples in five regular-size scenarios on three networks with number of paths range from 24–501. A large-scale problem with number of paths reaching 250,000 is visited for design of large-scale numerical study.

Figure **1-1** shows traffic network of Nguyen.

Figure **1-2** shows the network of Sioux Falls.

Figure **1-3** shows the network of Chicago Sketch.

These networks will be discussed as example to present the idea of our model.

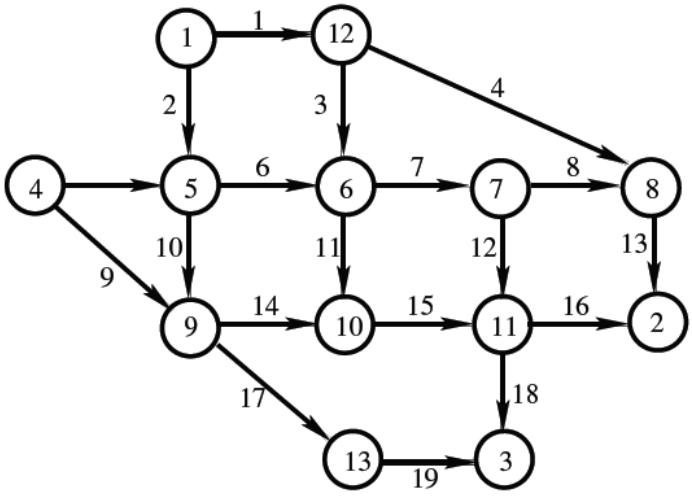

Figure **1-1** Network of Nguyen

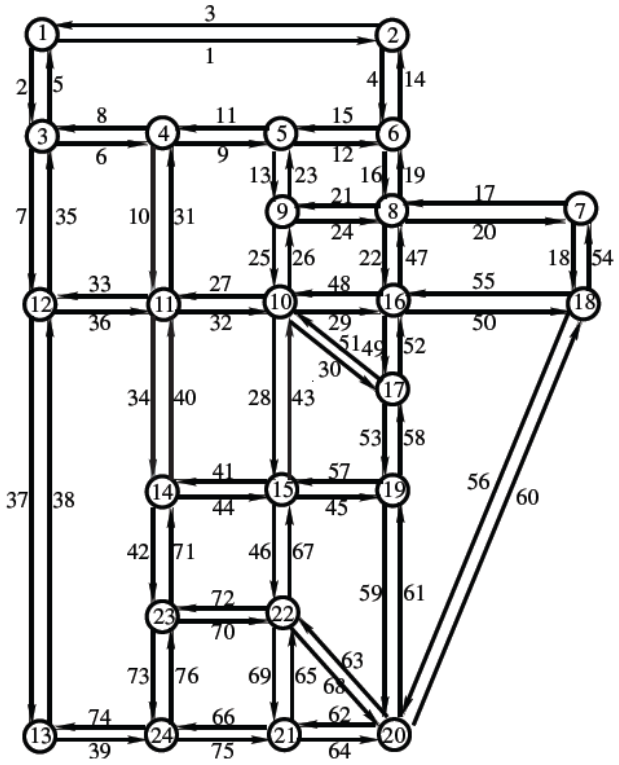

Figure **1-2** Network of Sioux Falls.

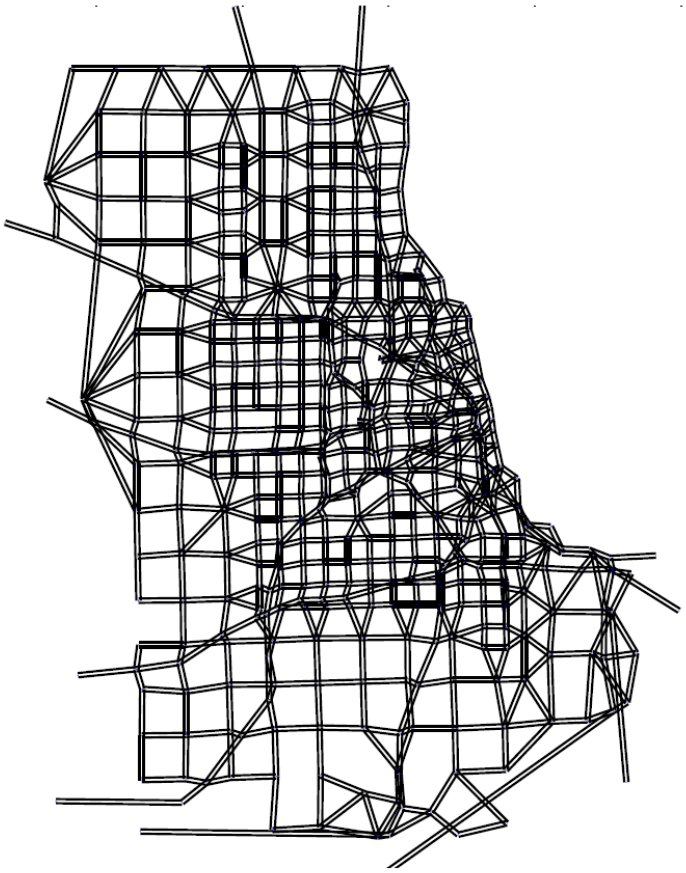

Figure **1-3** Network of Chicago Sketch

Notations

We introduce a few notations and terminologies for the ease of presentation as follows.

Chapter 2[6]	
$D_p(t,h)$	Delays on path p
$\Psi_p(t,h)$	Effective delay on path p
h	Departure rates
h^*	Equilibrium departure rates
Q_{ij}	Demand on OD pair (i,j)
$\mathcal{W}$	Set of OD pairs
p	Path
h_p	Departure rates on path p
t	Time
t_0	Starting time
t_f	Ending time
$\mathcal{P}$	Set of paths
n	Number of time intervals
Chapter 4	
Ψ	Delay operator
$\boldsymbol{h}$	Finite dimensional counterpart of h
$\boldsymbol{h}^*$	Finite dimensional counterpart of h^*
I_i	Interval i of the time horizon

[6] Some of the notation introduced in Chapter 2 is used throughout the rest of the dissertation.

i	Time index
w	Parameter in distance metric
d	"Distance" in Kriging
θ	Parameter in Kriging
Φ	Close form delay operator (equivalent to $\tilde{\Psi}$)
$\tilde{\Psi}$	Delay operator estimated by Kriging
δ_{pq}	Similarity between path p and path q
(r, s)	Origin–Destination (OD) pair
Λ	Feasible region for departure rates in continuous-time DUE
$\cdot$	Inner product
$\otimes$	Kronecker product
$\mathbb{R}$	Set of all real numbers
$\mathcal{C}$	Covariance function
Σ	Covariance matrix
Chapter 5	
ρ_{pq}	Shared-Link Similarity
τ_{pq}	Delay Similarity (DS)
$\tilde{\tau}_{pq}$	Effective Delay Similarity (EDS)
$Y(s;\ t)$	spatio-temporal random process
Chapter 6	
G	Number of aggregated paths
g	Aggregated path

$\mathcal{G}$	Set of all aggregated paths
Θ_g	delay on aggregated paths g
Chapter 7	
$D_p(t,h)$	True/exact delay operator
$\widetilde{D}_p(t,h)$	Metamodel (approximate delay operator)
$\widetilde{\Psi}_p(t,h)$	Surrogate/approximate effective delay operator

Chapter 2

Dynamic User Equilibrium (DUE)

The open-loop type of DUE, which is one type of DTA, is one of the most studied problems in modern traffic modeling. DUE is viewed by many scholars as a natural extension of Wardrop's first principle, which is used in traditional static traffic assignment for traffic networks. A traffic system is said to have reached dynamic user equilibrium (DUE) when unit travel costs, including early and late arrival penalties, are identical for all route- and departure-time choices selected by travelers between a given origin–destination (OD) pair (Han, 2013). That is to say, no traveler can reduce his or her travel cost by univariately change his or her travel decisions. DUE is a natural extension of the static user equilibrium model in which time is incorporated. Extensive literature on the DUE problem has been published. Two main variants of the DUE have been defined and studied in the related literature: the route-choice-only model (Friesz et al., 1989; Merchant and Nemhauser, 1978a, b; Mounce, 2006; Smith and Wisten, 1995; Zhu and Marcotte, 2000) and the simultaneous route and departure time choice model (Friesz et al., 1993, 2001, 2011, 2013a; Ran and Boyce, 1996; Wie et al., 2002).

A DUE model usually has five essential components (Peeta and Ziliaskopoulos, 2001):

1. a model of path delay,
2. flow dynamics,
3. flow propagation constraints,
4. a route- and departure-time choice model, and
5. a model of demand growth (evolution).

The DUE model seeks to mathematically describe and predict equilibrium route flow in a traffic system. Items 1, 2, and 3 are usually combined and referred to as the dynamic network

loading (DNL) subproblem, corresponding to Component (ii) of a DTA model[7]. The DNL subproblem will be introduce in detail in Chapter 4. Item 4 mathematically articulates the notion of DUE in computable form. It is the actual traffic assignment aspect of DUE modeling, corresponding to Component (i) in a DTA model. Item 5, the model of demand growth, occurs on a day-to-day time scale and allows travel demand to be updated.

DUE is conventionally studied as an open-loop, non-atomic Nash-like game (Friesz et al. 1993). The notion of open loop refers to the assumption that the travelers' route choices do not change in response to dynamic network conditions after they leave the origin. The non-atomic nature refers to the prevailing technique of flow-based modeling, instead of treating the traffic as individual vehicles. This is done in contrast to agent-based modeling (Balmer et al., 2004; Cetin et al., 2003; Shang et al., 2017, Han et al., 2019).

We introduced the Nash-like game concept to dynamic transportation models. Some formulation methods of a Nash-like DUE problem are listed below:

1. Variational inequalities (Friesz et al., 1993, 2013; Han et al., 2013b, 2015a, b; Smith and Wisten, 1994, 1995);

2. Nonlinear complementarity problems (Han et al., 2011; Pang et al., 2011; Ukkusuri et al., 2012; Wie et al., 2002);

3. Differential variational inequalities (Friesz and Meimand, 2014; Friesz and Mookherjee, 2006; Han et al., 2015a);

4. Differential complementarity systems (Ban et al., 2012);

5. Fixed-point problems in Hilbert spaces (Friesz et al., 2011; Han et al., 2015b); and

6. Stationary points of evolutionary dynamics (Mounce, 2006; Smith and Wisten, 1995).

[7] The two components in a DTA model are introduced in Chapter 1.

Most published mathematical formulations express an open-loop version of a dynamic notion of user equilibrium based on some type of generalization of Wardrop's first principle. (Friesz et al., 2018) There are many ways the DUE can be formulated. In this dissertation, we emphasize the VI formulation that was first introduced by Friesz et al. (1993) and subsequently widely used for DUE expressions. We adopt the VI formulation to discuss DUE and bi-level problems in subsequent chapters. The full VI formulation is presented in detail in the next section.

Over the past two decades, there have been many efforts to develop a theoretically-sound formulation of DUE that also has a canonical form that is acceptable to scholars and practitioners alike. Some models treat both route choice and departure time choice as fundamental decisions, while others are concerned with either departure time choice or route choice, but not both. (Friesz et al. 2018) Therefore, analytical DUE models developed involve the use of two varieties of user choice strategies:

- Route-Choice (RC) DUE
- Simultaneous Route-and-Departure-Time (SRDT) choice DUE

Details of these two types of DUE are introduced in subsequent sections of this Chapter.

In addition, it is to be emphasize again that dynamic user equilibrium (DUE) require some form of DNL or network performance model be carried out to determine the effective path delay. Since DNL remains to be the non-replaceable fundamental-level aspect of DUE, any analytical properties DNL does not enjoy are posing barriers in dynamic user equilibrium advances. Although some network performance models are extensively studied, many are not close-form and differentiable. In other word, in a DUE model, the analytical properties in the fundamental levels (e.g. the DNL model) are determining the properties and barriers in of the entire model.

Mathematical formulation of Dynamic User Equilibrium (DUE)

There are a few different ways a DUE may be formulated. Although some formulations are equivalent, it must be emphasized that all are not equivalent. (Friesz et al. 2018) In this subsection, we define the SRDT choice DUE and present its mathematical formulation. As introduced in the beginning of Chapter 2, a traffic system is said to have reached DUE when the experienced travel cost is identical for all the route- and departure-time choices selected by travelers between a given OD pair. The experienced travel cost includes travel-time and early/late arrival penalties, and is modeled by the effective delay operator.

We first define the effective delay operator for the SRDT choice DUE. We consider a fixed time horizon $[t_0, t_f] \in \mathbb{R}$. Because of the involvement of departure time choice, in this section, we distinguish between the delay operator and effective delay operator. We recall that the path delay operator provides the time to traverse any path (p) with any given departure time t, given a set of path departure rates, denoted by $h(\cdot) = h_p(\cdot): p \in \mathcal{P}$. In this chapter, the delay operator is denoted by

$$D_p(t, h) \; \forall \; p \in \mathcal{P}, \forall \; t \in \; [t_0, t_f \,]$$

where $\mathcal{P}$ is the set of paths used by travelers, t denotes the departure time, and h is a vector of departure rates. We stipulate that any path departure rate $h_p(\cdot)$ is nonnegative and square-integrable with respect to time t, and thus we have $h_p(\cdot) \in \; L_+^2[t_0, t_f \,]$.[8] Therefore, we have $h \in (L_+^2[t_0, t_f \,])^{|\mathcal{P}|}$.[9]

[8] $L_+^2[t_0, t_f]$ is the space of square-integrable functions defined on the interval $[t_0, t_f]$, and the subscript '+' indicates non-negativity.

[9] $(L_+^2[t_0, t_f])^{|\mathcal{P}|}$ is the $|\mathcal{P}|$-fold product of the space $L_+^2[t_0, t_f]$

The effective delay operator is then defined as a mapping between the set of path departure rates h and the set of experienced path travel times:

$$\Psi_p(t,h) \doteq D_p(t,h) + \mathcal{F}(t + D_p(t,h) - T_A)$$

where T_A is the predetermined target arrival time, and $\mathcal{F}(\cdot)$ is the penalty function each traveler is subjected to, associated with early or late arrival. $\Psi_p(t,h)$ can be interpreted as the perceived travel cost of travelers, in a dynamic setting, when he or she departs at time t and takes path p. The perceived travel cost includes the travel time, as given by the delay operator $D_p(t,h)$, which is similarly defined in the RC DUE, and an additional arrival time penalty term, which describes the cost a traveler endures when he or she arrives too early or too late from his or her planned arrival time.

For the formulation of SRDT choice DUE, we introduced the following constraints on h, usually know as demand satisfaction constraints or flow conservation constraints.

$$\sum_{p \in \mathcal{P}_{rs}} \int_{t_0}^{t_f} h_p(t)dt = Q_{rs} \quad \forall (r,s) \in \mathcal{W} \tag{2.1}$$

where W is the set of all OD pairs in the network, and $\mathcal{P}_{rs}$ is the set of paths connecting the OD pair $(r,s) \in W$. Q_{rs} is the fixed total travel demand for OD pair (r,s), given a priori.

With the formulation and notation introduced, we have the set of feasible solutions for the DUE problem:

$$\Lambda = \left\{ h \in \left(L_+^2[t_0,t_f]\right)^{|\mathcal{P}|} : \sum_{P \in \mathcal{P}_{rs}} \int_{t_0}^{t_f} h_p(t)dt = Q_{rs} \quad \forall (r,s) \in \mathcal{W} \right\}$$

Remark 2.1. There are two issues that we would like to clarify: (1) The SRDT DNL defined in this section is different from the RC DNL defined in Chapter 4, only regarding constraints on the vector of departures h. Therefore, we can define the finite dimensional counterpart with flexible

temporal granularity for the SRDT DNL in a similar way, which is omitted here. (2) In SRDT setting, the effective delay operator is introduced. However, the metamodeling will still be performed on the delay operator. Because the difference between the delay operator and the effective delay operator is the arrival penalty term, which usually can be computed fast and does not need to be involved in metamodeling. This is implying that the metamodeling task for both types of model (with and without departure time choice) are identical except for having different sampling regions, which can be adapted by using well designed sampling method.

Continuous-time SRDT DUE

Now, we are ready to mathematically formulate the continuous-time SRDT DUE. The notion and formulation are first introduced by (Friesz et al., 1993), who provide a definition tantamount to the following:

Definition 2.1. (Dynamic user equilibrium) A vector of departure rates $h^* \in \Lambda$ is a DUE if

$$h_p^*(t) > 0, p \in \mathcal{P}_{rs} \Rightarrow \Psi_p(t, h^*) = v_{rs} \in \mathbb{R}_{++} \, \forall (r,s) \in \mathcal{W}$$

The VI representation is presently the primary mathematical form employed for both route choice and SRDT choice DUE. The DUE defined in the equation above can be reformulated by a VI (Friesz et al., 1993) as follows:

$$\left.\begin{array}{c} find\ h^* \in \Lambda\ such\ that \\ \sum_{p \in \mathcal{P}} \int_{t_0}^{t_f} \Psi_p(t, h^*) \left(h_p(t) - h_p^*(t)\right) dt \geq 0 \\ \forall\ h \in \Lambda \end{array}\right\} VI\left(\Psi, \Lambda, \left[t_0, t_f\right]\right) \quad (2.2)$$

We induced a generalized form or inner product $\cdot$, which is defined as (Han et al. 2019)

$$\langle f, g \rangle \doteq \sum_{p \in \mathcal{P}} \int_{t_0}^{f_f} f_p(t) g_p(t) dt \quad \forall f, g \in \left(L^2\left[t_0, t_f\right]\right)^{|\mathcal{P}|}$$

This enables us to formulate DUE as

$$\langle \Psi(h^*), h - h^* \rangle \geq 0 \quad \forall h \in \Lambda$$

This is one version of the continuous time model presented in Friesz et al. (1993) and modified by various authors including Wie et al. (1995b), Friesz et al. (2011), Friesz et al. (2013), Han et al. (2013c), and Han et al. (2015b). Some other researchers, including Ran et al. (1993), Ran and Boyce (1996a), Ran and Boyce (1996b), Ran et al. (1996), also worked on continuous time DUE and created a family of DUE models sharing a common foundation. This is based on an equivalent optimal control formulation, which is used for showing the best responses of generic travelers constitutes a DUE flow pattern. These models replace the traditionally viewed unit effective travel delay operator, but embedded the notion of network loading within their DUE model. Such approach relies on very specific arc entrance and exit flow functions, as well as novel flow propagation constraints that are quite different than the continuous time physical identities used by Friesz et al. (2001), Perakis and Roels (2006), Kachani and Perakis (2006), and Kachani and Perakis (2009) (Friesz et al. 2018).

DUE in discrete time

DUE model can be presented in discrete time. This reformulation provide convenience for analytical solution schemes that are designed for finite dimensional models. Merchant and Nemhauser (1978a, b) set the stage for future work on DUE by using a discrete-time dynamic programming approach, solely with DSO traffic flow. Subsequent discrete-time DUE models include those by Drissi-Katouni and Hameda-Benchekroun (1992), Wie et al. (1995b), Huang and Lam (2002), and Nahapetyan and Lawphongpanich (2007). More recently Lo and Szeto (2002b) and Lo and Szeto (2004) have also made significant contributions to discrete-time DUE modeling with embedded CTM to accomplish DNL. In particular, their work confirms that LWR-based

DNL could be integrated with pure dynamic user equilibrium and, thereby, launched many investigations into LWR-based DNL/DUE. The authors cited for their work on discrete-time DUE used a variety of formulations including mixed integer linear programming, nonlinear programming, dynamic programming, optimal control theory, complementarity, and variational inequalities. In some instances, they were merely on departure time choice and in other instances on SRDT choice.

We may give the following discrete-time statement of DUE:

$$\left.\begin{array}{c} h_p^*(t_k) > 0, p \in \mathcal{P}_{ij} \Rightarrow \Psi_p(t_k, h^*) = v_{ij} \quad \forall (i,j) \in \mathcal{W} \\ v_{ij} = \min_{k,p \in \mathcal{P}_{ij}} \Psi_p(t_k, h^*) \ \ \forall (i,j) \in \mathcal{W} \\ \sum_{p \in \mathcal{P}_{ij}} \sum_{k=1}^{N} h_p^*(t_k) \cdot \Delta t = Q_{ij} \quad \forall (i,j) \in \mathcal{W} \\ \mathrm{h}^* \geq 0 \\ \Psi(t, h^*) = argDNL \end{array}\right\}$$

This discrete model retains all the properties of the original model.

Classifying DUE models

Route-choice (RC) only DUE

The RC model assume that the department time of each user is fixed. The user chooses route to take on. In contrast, with simultaneous route-and-departure-time setting, OD pairs are given.

Han et al. (2019) summarizes articles that addresses route choice DUE in recent two decades. These articles are: Bliemer and Bovy (2003); Chen and Hsueh (1998); Lo and Szeto (2002b); Long et al. (2013); Ran and Boyce (1996b); Tong and Wong (2000); Varia and Dhingra (2004); Zhu and Marcotte (2000).

Simultaneous route-and-departure-time (SRDT) choice DUE

Simultaneous route-and-departure and RC-time model is a major type of user choices model in DTA. The simultaneous route-and-departure-time (SRDT) choice DUE assumes each user to make simultaneously two decisions: on route and on departure time. The origin and destination are determined and pre-assumed by the set of introductions on simultaneous departure and RC model.

Challenges of the model is two-fold: modeling user behavior in time realistically, and assembling user's departure time choice with the entire model within the DUE framework. In introducing departure time choice, the model become deferent because now users' departure enjoy flexibility in time. This is also considered more realistic in describing road traffic. At the same time, when each user chooses both departure time and route to take, the decision is a dynamic one. Overall traffic status by this setting is expected to have more complexity and of high non-linearity. Gridlocks, traffic jams are integral to the definition of a dynamic user equilibrium and have to date been mainly expressed as variational inequalities, quasi-variational inequalities or complementarity problems. These expressions include discrete and continuous time. According to the summary in Han et al. 2019, SRDT choice DUE is discussed in the following works. (Friesz et al. 1993, 2001, 2011, 2013; Han et al. 2013b, 2015a, b; Huang and Lam 2002; Nie and Zhang 2010; Szeto and Lo 2004; Ukkusuri et al. 2012; Wie 2002). Emerging literature on DVI still to be exploited for modeling and computation.

Bounded rationality

We follow Friesz et al. (2018) for a discussion on bounded rationality. It is commonly known that bounded rationality plays an important role in modeling dynamic user equilibria

(Mahmassani and Chang, 1987; Mahmassani and Liu,1999). In recent two decades, some articles, include but not limited to: Mahmassani et al. 2005, Szeto and Lo 2006, Guo and Liu 2011, Get and Zhou 2012, Wu et al. 2013, Di et al. 2013, and Han et al. 2015b have analyzed bounded rationality for an array of timescales and notions of dynamic adjustment processes. The presence and style of expressing bounded rationality constitutes an additional consideration in classifying DUE models.

Elastic demand

Elastic demand has long been embraced as a critical component of all general static traffic assignment. For models with elastic demand, total demand on a network is a variable.

The discussion on elastic travel demand is important. Friesz et al. (1993) describe in prose how to extend the results for fixed trip tables to a new formulation employing inverse demand functions, in applying elastic travel demand models within a DUE modeling framework. Contribution to the study of DUE and DUE-like models and algorithms follows in subsequent years, including: Ran and Boyce (1996a), Cantarella (1997), Yang and Huang (1997), Yang and Meng (1998), Wie et al. (2002), Bliemer and Bovy (2003), Szeto and Lo (2004), and Bellei et al. (2005, 2006), and Han et al. (2011). The technical details of the model presented by Friesz et al. (1993) is first carried out by Friesz and Meimand (2014). Han et al. (2015a) extended the analysis into the dynamic and elastic demand dynamic user equilibrium (E-DUE) using functional analysis. The analytical properties and convergence of algorithms are also addressed in Han et al. (2015a).

Instantaneous/reactive DUE

Instantaneous or reactive DUE has been implemented in a number of analytical and simulation-based DUE models. These models, aside from a certain behavioral foundation, were mostly driven by the desire to avoid path enumeration and/or iterative computational procedure (especially given the high cost of implementing DNL procedures, e.g., microsimulation modeling). Some scholars who discuss these approaches include Li et al. (2000), Kuwahara and Akamatsu (1997), and Han (2003) (Friesz et al. 2018).

More recently Ma et al (2012a) have introduced a continuous-time model of so-called instantaneous dynamic user equilibrium that considers departure time choice but not simultaneous departure time and route choice. Moreover, the Ma et al (2012b) model employs a Vickrey-type DNL submodel. As we have commented previously, Vickrey-type network loading brings with its limitations on behavior, and does not reside at the current DNL research frontier. The Ma et al (2012b) instantaneous perspective ignores the critical notion of route choice and, as such, its solutions arguably do not constitute a user equilibrium. In fact, the Ma et al (2012b) instantaneous perspective has more in common with the early DTA literature on timing departure decisions (such as Friesz et al, 1989) than it does with the widely adopted notion of SRDT DUE. In subsequent papers Ban et al (2012b) and Ban et al (2014) do consider SRDT DUE but again through the prism of Vickrey-type DNL, preventing them from achieving the generality found in Friesz el (2013), Han et al (2015a) and Han et al (2015b) who use LWR-based DNL in conjunction with SRDT DUE.

Computation of DUE

Various solution schemes have been proposed for use in finding a solution to the DUE problem discussed in the previous subsection. Han et al. (2019) presented a table of computational algorithms developed for DUE. One of the frequently-employed methods is the fixed-point method. The fixed-point method is based on the differential variational inequality reformulation of DUE. The DVI formalism of DUE is derived by expressing the VI as a fictitious optimal control problem and then applying the minimum principle (Friesz et al., 2013a). Because we have limited concerning space, we do not include details about the reformulation and derivation of DVI. Friesz et al. (2013a) presents a detailed discussion of the DVI and the fixed-point method. The fixed-point method is elementary and attractive from coding point of view. It has been used successfully in several studies (Han et al. 2019).

Some other methods to solve the DUE problem include

- Conjugate gradient projection method (Lee et al. 2003)
- Nonlinear complementarity problem by successive linearization with Lemke's LCP algorithm (the PATH package)
- Gap function (Friesz et al. 2010)

In next subsection, we follow Han et al. (2015d) for the description of the fixed-point method.

Fixed-point computational scheme

Fixed-point method

Step 0 Identify an initial feasible point $h^0 \in \Lambda$. Set the iteration counter $k = 0$. Let $\alpha > 0$ be a fixed constant.

Step 1 Solve the DNL subproblem with path departure rates given by h^k, and obtain the path travel cost $\Psi_p(t, h^k), \forall\, p \in \mathcal{P}, \forall\, t \in [t_0, t_f]$.

Step 2 For each $(r, s) \in \mathcal{W}$, solve the following algebraic equation for μ_{rs}, using root search algorithms.

$$\sum_{p \in \mathcal{P}_{rs}} \int_{t_0}^{t_f} \left[h_p^k(t) - \alpha \Psi_p(t, h^k) + \mu_{rs}\right]_+ dt = Q_{rs}$$

Then update the next iterate $h^{k+1}(\cdot) = \left\{h_p^{k+1}(\cdot) : p \in \mathcal{P}\right\}$ where

$$h_p^{k+1}(t) = \left[h_p^k(t) - \alpha \Psi_p(t, h^k) + \mu_{rs}\right]_+ \quad \forall p \in \mathcal{P}_{rs},\ \ (r, s) \in \mathcal{W}$$

where $[x]_+ \doteq \max\{0, x\}$.

Step 3 Terminate the algorithm with output $h^* \approx h^k$ if

$$\left\|h^{k+1} - h^k\right\|_{L^2} / \left\|h^k\right\|_{L^2} \leq \epsilon$$

where $\epsilon \in \mathbb{R}_{++}$ is a prescribed termination threshold, and the norm $\|\cdot\|_{L^2}$ is defined as

$$\|h\|_{L^2} = \left(\sum_{p \in \mathcal{P}} \int_{t_0}^{t_f} \left(h_p(t)\right)^2 dt\right)^{1/2}$$

Otherwise, set $k = k + 1$ and repeat Step 1 through Step 3.

DTA simulation

Many of the early DTA simulation models take time-dependent OD matrices as inputs and equilibrate time-dependent route shows using a static traffic assignment for each given time step of a DTA (Peeta and Ziliaskopoulos 2001). Such crude models do not account for the formation of demand and usually restrict themselves to route choice. That is to say, they offer no behavioral basis for departure time choice (Nagel and Flötteröd, 2012) and ignore the necessity of

simultaneous route choice and departure-time choice that ensures the intrinsic consistency needed to identify DUE solutions. Modern DTA simulation and computer packages provide new resources for DTA research.

TRANSIMS is a package that estimates second-by-second movements of individual travelers while exploiting parallel processing and cellular automata representations. Jeihani (2010) reviews selected well-known computer packages including TRANSIMS. Using TRANSIMS is observed to address some of the existing problems in DTA models. (Friesz et al. 2018).

Agent-based simulation (ABS) is another simulation choice alternative to regular DTA simulation. Rules, equations, and inequalities are used to capture the behavior of individual road network users. ABS has made great progress in recent years and become a largely accepted tool for capturing DSO and DUE flow patterns. Successful ABS models offer researchers a convincing way to represent behavioral subtleties. As pointed out by Nagel and Flötteröd (2012), ABS has the particular property of directly representing the feedback from changing network conditions to user decisions. Also, it is worth pointing out that ABS explicitly model departure time choice, mode choice, and activity choice of users. In other word, the user behaviors are explicit. ABS models may be crafted to include the implications of constraints intrinsic to differential algebraic equation (DAE) systems used for DNL modeling (Friesz et al. 2018).

Other important simulation packages include: 1) DYNA-SMART: a discrete time mesoscopic simulation model for transportation systems. It is designed to model traffic pattern and evaluate overall network performance under real-time information systems[10]; and MATSIM: development of this package is started by a team led by Kai Nagel that does microscopic traffic

[10] This brief introduction follows its website http://www.its.uci.edu/ctss/sim_models/dynasmart.html. For more information on this package, the readers are referred to the webpage.

simulation. MATSim is an activity-based, extendable, multi-agent simulation framework implemented in Java. It is open-source. The framework is designed for large-scale scenarios. This means all models' features are stripped down to efficiently handle the targeted functionality. For the network loading simulation, MATSim implemented a queue-based model and omit very complex and computationally expensive car-following behavior (Horn et al. 2016, Axhausen et al. 2016).

Chapter 3

Kriging/Metamodeling

History of Kriging

Statistical learning (machine learning) has enjoyed increasing popularity in academia, industry, and social media in recent years. In this dissertation, we utilize Kriging, a statistical learning method that originates with gold mining in South Africa. Developed in 1960s by Krige, the method grows beyond its original application field, and becomes popular in computer experiments in 1990s when the calculation power of computers had its significant growth. In the entire pool of all machine learning methods, Kriging enjoys unique quality of flexibility and accuracy that is not shared with other popular machine learning method, which is discussed in subsequent sections.

Kriging involves the idea of modeling a target model or operator by a realization of a random process. Statistical and analytical nature of the ensemble[11] of the realizations is the foundation of the flexibility and modeling strength it provided. Here by "ensemble", we are referring to the large set of (equivalent) realizations from a random process. Kriging models a target mapping by one realization from the ensemble of a Gaussian process. Due to the use of realizations of Gaussian process, which are highly flexible, Kriging is significant in being sufficiently flexible to create metamodel for non-differentiable, nonlinear models. The procedure of using Kriging to model a target function to a smooth "surface" is called response surface methodology.

[11] The word ensemble may refer to a concept in physics that has now been widely adopted in other fields. Here we use the word in the context of discussing within the statistics field.

Categories of metamodeling

Gaussian process

In this section, we start with a definition of Gaussian process follows Fang et al. (2005):

Definition 3.1 (Gaussian process) A stochastic process $\{X(t), t \in T\}$ indexed by t is said to be a *Gaussian process* if any of its finite dimensional marginal distribution is a normal distribution, i.e., if for any finite set $\{t_1, \dots, t_p\}$, $(X(t_1), \cdots, X(t_p))$ has an p-dimensional normal distribution.

We denote the mean, variance, and covariance functions by $\mu(t) = E\{Y(t)\}, \sigma^2(t) = Var\{Y(t)\}$, and $\sigma(t_1, t_2) = Cov\big(X(t_1), X(t_2)\big)$ for any $t_1, t_2 \in T$. A Gaussian process $\{X(t), t \in T\}$ indexed by t is said to be *stationary* if its $\mu(t)$ and $\sigma^2(t)$ are constant (independent of the index *t*), and its $Cov\big(X(t_1), X(t_2)\big)$ depends only on $|t_{i1} - t_{i2}|, i = 1, \dots, p$, where $t_j = \big(t_{1j}, \dots, t_{pj}\big), j = 1,2$.

In our study, Kriging is one of the main metamodeling methods we use. Kriging considers the observed input-output functional relationship as a realization of a Gaussian Random Field (GRF), and the resulting predictor corresponds to the mean of the posterior predictive density of the function approximation. It is to be noted that Kriging does not assume that the target model follows Gaussian distribution in any sense. Also, Kriging does not require any distribution assumptions on the model to be learned. Taking the advantages of the properties of a GRF enhance the model by allowing us to obtain a posterior distribution which we can use to predict. The mean of this distribution coincides with the kriging predictor, and the variance expression fully characterizes the posterior distribution. This variance is shown in Figure 3-1 as the "width" of the confidence band (represented by grey shadowed area).

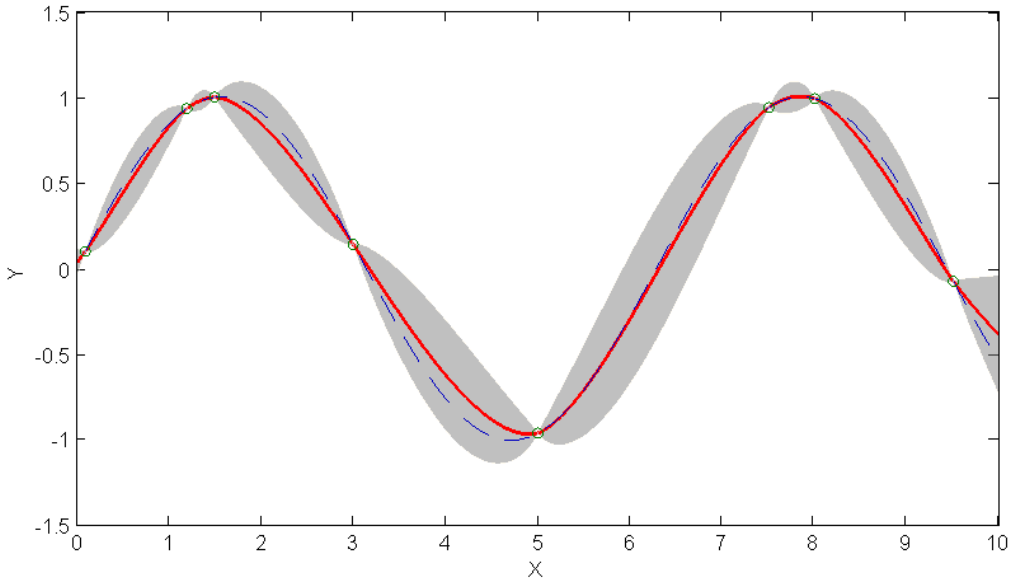

Figure **3-1** Illustration of Kriging

The confidence bands in Figure 3-1 is drawn according to the variance of the predictor from the posterior distribution induced by GRF. The shape of the confidence band illustrates a very important property of Kriging: its predictions on the samples (points in the training dataset) are exact. This directly makes the variance goes to zero at the sample points represented in green circles. When the original model to be learned is very expensive or time consuming, as it is in our case, this property is preferable in the sense that the metamodel exactly mimics the original model where original model had already been evaluated. The importance of this property on our developed metamodeling scheme will be furtherly discussed in Chapter 4.

Kriging

There are several categories of Kriging to satisfy a range of modeling needs. An integral piece of the model is the covariance function selected. In this section, we discuss spatio Kriging

to illustrate the idea. Discussion on covariance functions of spatio-temporal is separately given in subsequent section.

To determine spatio covariance function $C(\cdot)$, Kriging assumes that $Y(x)$ satisfies second-order stationary, which means that its expectation and variance do not depend on actual locations, but only on the distance d between $x + d$ and x (Cressie 1993):

$$E[Y(x+d) - Y(x)] = 0 \quad \forall d \in \mathbb{R}, x \in \mathbb{R}^2$$

$$Var[Y(x+d), Y(x)] = C(d) \quad \forall d \in \mathbb{R}, x \in \mathbb{R}^2$$

These equations are defined on the two-dimensional input space, but can be extended to accommodate $x \in \mathbb{R}^n$ without difficulty. Because the variance of Y^* must be nonnegative, it can be concluded from definition of Kriging that (Cressie 1993; Curriero 2006; Zou et al. 2012)

$$Var(Y^*) = \sum_i \sum_j \lambda_i \lambda_j C(h) \geq 0$$

This is called conditional positive-definite, which is of vital importance in determining the validity of the spatial covariance function. A large body of researchers proposes various valid spatial covariance functions, including exponential, Gaussian, spherical, and Matern (e.g., Cressie 1993, Diggle and Ribeiro 2007). All these classical spatial covariance functions are based on Euclidean distance.

Tradition metric, however, can be invalid in modeling special problems where the variable carries similarity on a non-Euclidean space (Zou et al., 2012). Therefore, Kriging based on the Euclidean distance metric may not be directly achieved, and a new distance metric or a new covariance function form should be proposed to solve problems having this nature. (Zou et al., 2012).

The following figure shows how Kriging learns a mapping from X to Y:

Figure **3-2** Illustration of Kriging on an $X \rightarrow Y$ mapping

The following sections respectively introduce common categories of Kriging.

Ordinary Kriging

Ordinary Kriging is the most commonly used Kriging model in practice (Fang et al., 2005). The model looks like follows:

$$y(\boldsymbol{x}) = \mu + z(\boldsymbol{x})$$

Ordinary Kriging uses weighted averages of the sample data to generate the interpolations. In the equation above, μ is a constant mean and $z(x)$ is a zero-mean Gaussian random process. For a two-dimensional spatial data, assume that sample data $\{Y(x_i): x_i \in \mathbb{R}^2\}$ are collected at sites $i = 1, 2, \ldots, n$ with locations x_i, then the unknown value Y^* in location x^* can be described by a linear combination of Y(x_i): (Zou et al., 2012)

$$Y^*(x^*) = \sum_{i=1}^{n} \lambda_i Y(x_i)$$

It is to be noted that Ordinary Kriging assumes no distribution. The prediction $Y^*(x^*)$ Kriging obtains should be seen as a point estimator. It is not a random variable following any assumed distribution, since no distribution is assumed. Kriging on higher dimensional space follows same structure of generating the interpolation (metamodel predictions) by linear combination of the sample observations.

Universal Kriging

The model of Universal Kriging is given as follows

$$y(x) = \sum_{j=0}^{L} \beta_j B_j(x) + z(x)$$

$$r(\boldsymbol{\theta}; \boldsymbol{x_1}, \boldsymbol{x_2}) = Corr(z(\boldsymbol{x_1}), z(\boldsymbol{x_2}))$$

where z(x) is a Gaussian process with zero mean, variance σ^2, and correlation function $r(\boldsymbol{\theta}; \boldsymbol{x_1}, \boldsymbol{x_2})$.

Spatio-temporal Kriging

Spatial temporal statistical model tries to model how current process values evolves from the past process values in a spatial process. The book by Cressie and Wikle (2011) provide a clear and comprehensive introduction on this area of research methods. To introduce this approach, we consider a spatio-temporal random process $Y(s;\ t), s \in\ D_s, t \in\ D_t$ that evolves through the spatio-temporal index set $D_s\ \times\ D_t$ following (Cressie and Wikle, 2011):

$$Y(s;t)\ = \mu(s;t) + \beta(s) + \gamma(t) + \kappa(s;t) + \delta(s;t), s \in D_s, t \in D_t$$

where $\mu(s;\ t)$ is a deterministic mean, $\beta(s)$ is a mean-zero random effect representing the location-specific variability common to all times, $\gamma(t)$ is a mean-zero random effect representing time-specific variability common to all locations, $\kappa(s;\ t)$ is a mean-zero random effect capturing the spatio-temporal interaction not found in the deterministic mean $\mu(s;\ t)$, and $\delta(s;\ t)$ is a mean-zero random effect modeling the white noise. The random effects are assumed to be mutually statistically independent. For some cases, the noise (or nugget) can be removed. In the next few subsections, we discuss theoretical properties of spatio-temporal Kriging.

Other metamodeling methods

Neural Networks (NN)

The term Neural Network has evolved to encompass a large class of models and learning methods in the last decade and it has become one of the most well-known-to-public statistical learning method (Hassoun (1995), Bishop (1995), Haykin (1998), Fang et al. (2005) and Hagan et al. (1996)). Here, we see "learning" as parameter estimation. Neural Networks are computer models inspired by biological structure of human brain to mimic highly intelligent cognitive functions. Neural networks in statistical metamodeling composed of simple elements (artificial neurons) operating in parallel based on a network function that is determined largely by the connections between elements (Fang et al .2005). We can train a Neural Network to perform a particular function by adjusting the values of the weights on the connections (i.e., parameters) between elements (the neurons). The input-output mapping can be viewed as a non-parametric regression computation (Fang et al .2005).

Radial basis function

Radial basis function (RBF) methods are techniques for exact interpolation of data in multi-dimensional space (Powell 1987). The RBF maps the inputs to the outputs using a linear combination of the basis functions (Fang et al. 2005). A number of modifications to the exact interpolation have been proposed (Moody and Darken (1989), Bishop (1995), Fang et al. 2005).

When Gaussian basis functions are chosen, the RBF model is closely related to the Gaussian Kriging model (Li and Sudjianto 2005, Fang et al. 2005).[12]

Other metamodeling methods includes support vector machines, smoothing splines and local polynomial regression are beyond the scope of this dissertation. Interested readers are recommended to consult the book by Fang et al. (2005) for references.

Metamodel Properties

Separability

Definition 3.2.(Cressie 2011) A random process $Y(\cdot;\cdot)$ is said to have a separable spatio-temporal covariance function if for all $\boldsymbol{s}, \boldsymbol{x} \in \mathbb{R}^d, t, r \in \mathbb{R}$ we obtain

$$cov(Y(\boldsymbol{s};t), Y(\boldsymbol{x};r)) = C^{(s)}(\boldsymbol{s},\boldsymbol{x}) \cdot C^{(t)}(t,r) \tag{3.1}$$

where $C^{(s)}$ and $C^{(t)}$ are spatial and temporal covariance functions, respectively. As a consequence of Equation (3.1), a simple class of spatio-temporal covariance functions is given by the product of individual spatial and temporal covariance functions.

[12] A Gaussian process with radial basis covariance is identical to an RBF model. Interested reader are referred to Cressie (2006) for more details.

Stationarity in space or time

Definition 3.3. We say that f is a stationary spatio-temporal covariance function on $\mathbb{R}^d \times \mathbb{R}$, if it satisfies nonnegative-definite[13] requirement and can be written as

$$f\big((\boldsymbol{s};t),(\boldsymbol{x};r)\big) = C(\boldsymbol{s}-\boldsymbol{x};t-r), \quad s,x \in \mathbb{R}^d \; t,r \in \mathbb{R}$$

If a random process $Y(\cdot;\cdot)$ has a constant expectation and a stationary covariance function, $C_Y(h;\ \tau)$, then it is said to be second-order (or weakly) stationary.

When $C^{(s)}$ and $C^{(t)}$ are, respectively spatially and temporally stationary, then the covariance function (3.1) becomes (Cressie, 2011)

$$C(h;\tau) = C^{(s)}(h) \cdot C^{(t)}(\tau), \;\; h \in \mathbb{R}^d, \tau \in \mathbb{R}$$

The concept of strong stationarity corresponds to the equivalence of the two probability measures defining the random processes $Y(\cdot;\cdot)$ and $Y(\cdot+h;\cdot+\tau)$, respectively, for all $h \in \mathbb{R}^d$ and all $\tau \in \mathbb{R}$.

Fully symmetry

Definition 3.4.(Cressie, 2011) A random process $Y(\cdot;\cdot)$ is said to have a fully symmetric spatio-temporal covariance function if

$$cov\big(Y(s;t),Y(x;r)\big) = cov(Y(s;r),Y(x;t))$$

[13] A covariance function is a nonnegative-definite function, and vice versa. See Cressie (2011) for the definition of nonnegative-definite function $\{f(u,v) : u,v \in D\}$ defined on $D \times D$

Metamodeling on Dynamic Network Loading (DNL)

In this dissertation, we employ Kriging method onto metamodeling the dynamic network loading (DNL) problem. We recall the ordinary Kriging model:

$$Y = \mu + \epsilon(x)$$

The model is built as follows

$$\Psi(\boldsymbol{h}) = \mu + \epsilon(\boldsymbol{h})$$

in which definition of the terms are:

$$\Psi: estimated\ delays$$

$$\boldsymbol{h}: departure\ rates$$

$$\epsilon: zero - mean\ Gaussian\ process$$

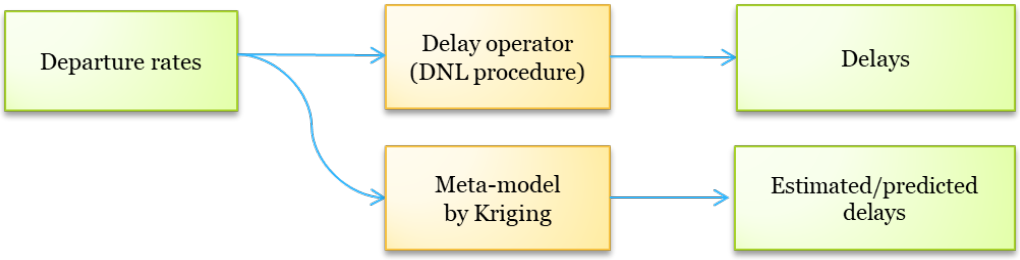

Figure **3-3** Metamodeling on delay operator

where $\boldsymbol{h}$ is an n × $|\mathcal{P}|$-dimensional vector. This framework follows general concept of metamodeling. By slight modification it can reflect the framework of metamodeling by other comparing methods. The only change needs to be made is to replace "meta-model by Kriging" component with a different metamodel component. Kriging enjoys high and resilient smoothness against practical non-smoothness when gridlock, spillback, and jams appears. Kriging on DNL is the foundation of our proposed extensions in Chapter 5, and bi-level problems in Chapter 7. The study including numerical tests on this approach is discussed in full details in Chapter 4.

Chapter 4

Statistical metamodeling on DNL

This chapter contains a discussion of statistical metamodeling applied to the dynamic network loading (DNL) aspect of DUE.[14]

Metamodeling is the process of building a "model of models," i.e., a surrogate approximation of the exact models/processes to improve the computation efficiency or gain better analytical properties (Wang and Shan, 2007). Specifically, we treat the delay operator as a highly nonlinear mapping from the set of path departure rates to the set of path travel times, and interpret and approximate its inherent input-output mechanism using Kriging. Kriging considers the observed $h - \Psi$ functional relationship as a realization of a GRF, and the resulting estimation corresponds to the posterior predictive density of the function approximation. The proposed framework is general enough to handle a wide range of DNL models with different traffic flow dynamics (i.e., it is a "model of DNL models"). Each output of this metamodel is an approximate DNL model with closed-form expression and superior regularity and computational efficiency.

- Kriging on DNL

Statistical/machine learning algorithms learn from and make predictions based on data, sometimes without exploration of the behavioral foundations and plausibility of the learning processes on which they are based. However, this is not the case in our proposed Kriging framework. Firstly, instead of making unstructured interpretations of the input-output mechanisms, the proposed approach utilizes information about the network structure, path, and time to identify parts of the input variables that are likely to be correlated, and defines the correlation functions accordingly. Secondly, although the path delay operators may be

[14] Some of the content in this section was previous presented in the article Song et al. (2018).

discontinuous, it is understood that the congestion effect, observed at some point in the spatio-temporal domain, tend to progress in a continuous way in space and time due to the finite propagation speeds of traffic characteristics (e.g., kinematic waves). The potential discontinuity of path delay operators may be dealt with in Kriging by making appropriate regions of the spatio-temporal surface very steep yet smooth.

There is a large and rapidly growing literature on machine learning methods (Hastie et al., 2009; James et al., 2013; Murphy, 2012). Among the many choices of statistical/machine learning techniques that can be used for metamodeling, Kriging is the only technique that has all the following three properties:

• It is an exact interpolator. This means that when approximating a function $Y(x)$ with a metamodel $\hat{Y}(x)$ at a data point x used to fit the model (a point in the training dataset), $\hat{Y}(x) = Y(x)$ (i.e., the prediction error at x is zero). This is an important property in metamodeling if running the original model to be approximated is very expensive or time consuming. It is, therefore, convenient that the metamodel exactly mimics the original model in instances in which the original model had already been ran. This property is not shared by other popular techniques such as support vector regression (SVR), NN, or Random Forest (RF);

• It provides a predictor that is closed-form and analytic. This is a property shared by Kriging and SVRs, but not by NNs or RFs. A closed-form Kriging predictor has a number of advantages in DTA applications as we discuss in both Chapter 3 and the last section of this Chapter;

• It provides a closed-form, analytical expression of the prediction error variance (or standard error). This property is exclusive to Kriging thanks to the interpretation of its predictor as the mean of the predictive posterior density of a Gaussian process (or GRF). The underlying Gaussianity assumption is used only if prediction intervals are desired, in which case Kriging provides minimum mean square error predictions (Santner et al., 2013). If the Gaussianity

assumption does not hold, then the Kriging predictor can be shown to be the best linear unbiased predictor (Santner et al., 2013).

It is worth mentioning here that there is an expression for variance of the posterior in Kriging. This variance could be used to determine how "narrow" the predictions will get as the size of training set increases, and deduce an error bound on the accuracy of the predictions and give references to support evaluation and prediction of surrogate model performances. This property is also not shared by many other metamodeling methods. Kriging is very promising in the group of machine learning methods. Judging by the result in our examples, Kriging performs great on transportation models given. For a spectrum of complex original problem, Kriging reduce its formulation to a conceptually simple one.

To the best of our knowledge, Kriging has not been used to model DNL before in literature. We are the first explorer to pursue the direction.

Metamodeling in DNL

Dynamic network loading and the delay operator

The notion of DNL varies in context and application in the literature. In this section we articulate the delay operator that this study addresses using precise mathematical languages while referring the reader to a number of other papers that discuss the DNL procedures as well as their numerical implementations in a wider spectrum (Friesz et al., 2013a; Gentile et al., 2007; Han et al., 2015b; Huang and Lam, 2002; Ukkusuri et al., 2012). However, we note that there is no fundamental difficulty to apply the proposed metamodeling framework to other types of DNL models and, in general, traffic network performance models.

Delay operator as an infinite-dimensional mapping

We consider a general network with a time horizon $[t_0, t_f]$. Let $\mathcal{P}$ be the set of all paths employed by travelers. For each $p \in \mathcal{P}$, we define its path departure rate $h_p(\cdot)$ as a function of departure time *t*. Then we let $h(\cdot) = h_p(\cdot)$: $p \in \mathcal{P}$ be the vector of path departure rates. The following constraints on the departure rates are commonly employed for RC DUE problems (Smith and Wisten, 1995; Zhu and Marcotte, 2000):

$$\Sigma_{p \in \mathcal{P}_{rs}} h_p(t) = R_{rs}(t) \qquad \forall t \in [t_0, t_f], \ \forall (r,s) \in \mathcal{W}$$

where $\mathcal{W}$ is the set of OD pairs, $\mathcal{P}_{rs}$ is the set of paths that connect OD pair $(r,s) \in \mathcal{W}$. $R_{rs}(t)$ is the time-varying departure rate between OD pair (r,s) (over all possible paths), and is given a priori. This constraint is commonly employed for RC DUE. We note that this is different from the constraints presented in formula (2.1). The fundamental difference is that in RC DUE, the departure rate between a given OD pair is fixed for each time instance, i.e., the travelers only decide about the route to take, but not when to go. In SRDT choice DUE, only the total demand between an OD pair is fixed, aligned with the assumption that travelers make both route-and-departure-time choice.

Therefore, the set of feasible path departure rates can be defined as

$$\Lambda = \left\{ h(\cdot): \ h_p(t) \geq 0, \ \sum_{p \in \mathcal{P}_{rs}} h_p(t) = R_{rs}(t) \qquad \forall t \in [t_0, t_f], \ \forall (r,s) \in \mathcal{W} \right\} \tag{4.1}$$

The delay operator is a mapping that relates the set of path departure rates h to the set of path travel times Ψ:

$$\Psi(h) \doteq \left(\Psi_p(t;h): \ t \in [t_0, t_f], p \in \mathcal{P}\right) \quad h \in \Lambda$$

where each $\Psi_p(t; \ h)$ denotes the path travel-time experienced by drivers departing at time t and following path p, when the path departure rates of the entire network are given by h. In

summary, we define the delay operator, which is viewed as an infinite-dimensional mapping, as follows.

Definition 4.1. (Infinite-dimensional delay operator) Given a road network and the feasible path flow set (4.1), a delay operator is a mapping from the set of path departure rates $h \in \Lambda$ to the set of path travel times $\Psi(h)$.

Remark 4.1. There are many different ways to perform the DNL procedure to evaluate the delay operator, including those mentioned in Chapter 1. Each way can be seen as a DNL model and hence there are many different DNL models. We propose one metamodel that can be applied to these individual DNL models by following the same procedure, which is to be elaborated in the sections below.

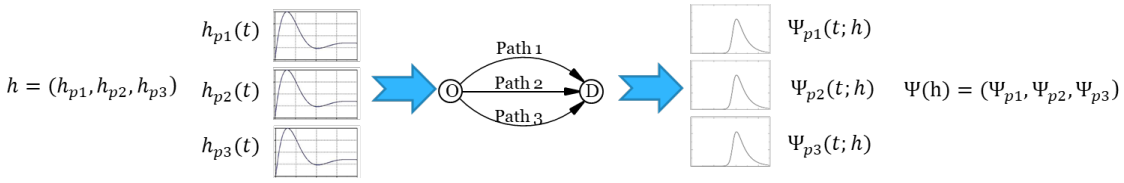

Figure **4-1** Illustration of the DNL model

Delay operator as a finite dimensional mapping

To apply the Kriging technique and facilitate numerical implementation, we need to define the delay operator in a finite dimensional space. To this end, we let n be an arbitrary positive integer, and partition the time horizon $[t_0, t_f]$ into n equal sub-intervals, denoted $I_i, i = 1, \ldots, n$. We define the discrete path departure rate to be an n-dimensional vector $\boldsymbol{h}_p \in \mathbb{R}_+^n$, and the entire vector of path departure rates to be a n × $|\mathcal{P}|$-dimensional vector $\boldsymbol{h} = (\boldsymbol{h}_p : p \in \mathcal{P}) \in \mathbb{R}_+^{n \times |\mathcal{P}|}$. To be clearer on variable dimensionality, starting from this section, vectors are written in bold and numbers/scalars are written in non-bold styles.

We then define the corresponding continuous-time path departure rates as follows:

$h(\cdot) = h_p(\cdot): p \in \mathcal{P}$ such that $h_p(t) \equiv \boldsymbol{h}_{p,i}\ t \in I_i, \forall 1 < i < n, \forall p \in \mathcal{P}$

h can be viewed as the continuous-time counterpart of $\boldsymbol{h}$. We then construct the mapping $\boldsymbol{\Psi}$ as follows:

$$\boldsymbol{\Psi}(\boldsymbol{h}) = \left(\boldsymbol{\Psi}_{p,i}(\boldsymbol{h}):\ \ p \in \mathcal{P},\ \ i = 1, \ldots, n\right) \in \mathbb{R}^{n \times |\mathcal{P}|}$$

where

$$\boldsymbol{\Psi}_{\boldsymbol{p,i}}(\boldsymbol{h}) \doteq \frac{1}{|I_i|} \int_{I_i} \Psi_p\,(t;\ h) dt$$

can be interpreted as the average OD departure rate during the $i - th$ time interval.

Definition 4.2 (Finite dimensional delay operator) Given a network and the feasible path flow set (4.1), a delay operator Ψ is a mapping from the set of discrete path departure rates h to the set of discrete path travel times $\Psi(h)$. As such, $\boldsymbol{\Psi}$ is a mapping between two subsets of the Euclidean space $\mathbb{R}^{n \times |\mathcal{P}|}$, and is viewed as the discrete counterpart of the delay operator Ψ.

Remark 4.2. The operator Ψ should not be confused with the delay operators with numerical discretization, such as those based on DNL performed via the cell transmission model or the link transmission model. Here, the selection of n is arbitrary, not constrained by the network or the time horizon, nor by the numerical discretization scheme (e.g. the Courant-Friedrichs-Lewy condition). The introduction of h and Ψ enables us to effectively trade temporal granularity for dimensionality. Moreover, a coarse time grid (small n) also makes sense in practice: in an actual network the average path departure rates or path travel times may not vary much over a time period (e.g. 30 min) that is significantly longer than a typical time step in numerical computations (e.g. 5s, 1 min). Thus, the delay operator with a small n is useful for provide average travel times within, say, 30 min.

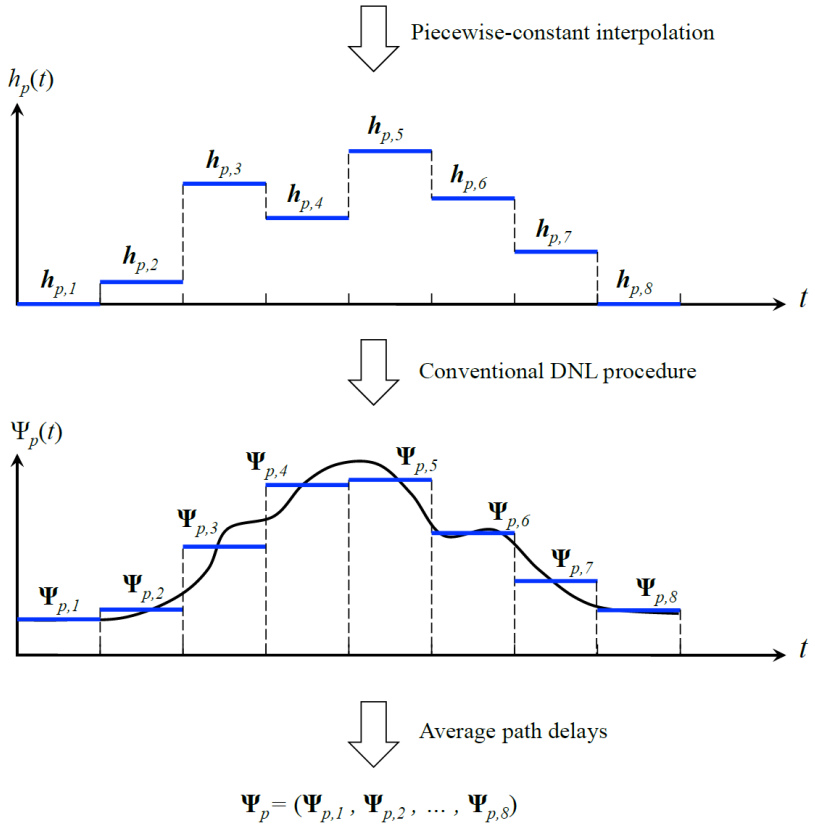

Figure **4-2** Illustration of the mapping from h_p to Ψ_p when $n = 8$, which defines the finite dimensional operator Ψ

The procedure above defines the delay operator in a finite dimensional setting, where the vector of path departure rates is represented by the vector h and the corresponding vector of path travel times is $\Psi(h)$. This process is graphically illustrated in Figure 3-1. The definition of allows us to apply Kriging techniques in the finite dimensional space $\mathbb{R}^{n\times|\mathcal{P}|}$.

Methodology-metamodel and experimental design

We propose the learning method based on Kriging and make statistical assumptions on the underlying delay operator Ψ to enable the analysis based in GRF. Based on the interdependency across all paths and time steps. In particular, the delays for a set of departure

choice pairs (p, i) can be considered as a realization of a Gaussian process, with the dissimilarity between any two such pairs defined as a vector of aggregated differences in the path departure rates, which we will introduce in detail in subsequent sections. We apply the *Ordinary Kriging* model, which is the most commonly used Kriging model in practice (Fang et al., 2005). For any vector of path departure rates $\boldsymbol{h} \in \mathbb{R}_{+}^{n\times|\mathcal{P}|}$, we assume that the mapping $\boldsymbol{\Psi}$ can be approximated by a realization of the random process:

$$\widetilde{\boldsymbol{\Psi}}_{p,i}(\boldsymbol{h}) = \mu_{p,i} + \varepsilon_{p,i}(\boldsymbol{h}) \qquad \forall (p,i),\ \ \forall \boldsymbol{h} \in \mathbb{R}^{n\times|\mathcal{P}|}$$

For each pair (p, i), $\mu_{p,i}$ is the deterministic mean, $\varepsilon_{p,i}(\cdot)$ is a Gaussian process with zero mean and covariance as follows:

$$cov\left(\varepsilon_{p,i}(\boldsymbol{h}^1), \varepsilon_{p,i}(\boldsymbol{h}^2)\right) = \mathcal{C}_{p,i}\left(\boldsymbol{h}^1, \boldsymbol{h}^2; \boldsymbol{\theta}^{p,i}\right) \qquad \forall (p,i)$$

where $\boldsymbol{\theta}^{p,i}$ is some vector of parameters to be estimated later. We further assume the stationary property. That is

$$\mathcal{C}_{p,i}(\boldsymbol{h}^1, \boldsymbol{h}^2; \boldsymbol{\theta}^{p,i}) = \tilde{\mathcal{C}}_{p,i}(\boldsymbol{d_i}(\boldsymbol{h}^1, \boldsymbol{h}^2); \boldsymbol{\theta}^{p,i}) \tag{4.2}$$

where $\boldsymbol{d}_i(\boldsymbol{h}^1, \boldsymbol{h}^2)$ is some dissimilarity indicator between $\boldsymbol{h}^1$ and $\boldsymbol{h}^2$.

Remark 4.3. The stationarity is a common assumption made to characterize the class of GRFs (Gaussian processes) we are considering. Without it, the statistical inference about the underlying probability law would be almost impossible based on a single realization of the field (Cressie and Wikle, 2011; Fang et al., 2005). Assuming stationarity means that the probabilistic structure of the random process looks similar in different areas of the experimental region (Santner et al., 2013). However, this is not a strong assumption because the GRF remains sufficiently flexible through anisotropic variance function and unknown parameters.

It is crucial to determine an appropriate form of the similarity function for the statistical model to perform properly. Here, we propose a dissimilarity function that uses different kernels

for different paths and time intervals. We first use δ_{pq} defined below to indicate the proximity or similarity between an arbitrary pair of paths p and q:

$$\delta_{pq} = \frac{\#\ of\ shared\ links\ between\ path\ p\ and\ path\ q}{average\ \#\ of shared\ links\ between\ p\ and\ all\ paths\ q'\in\ P} \tag{4.3}$$

which gives rise to the proximity matrix $\{\delta_{pq}\}\ where\ p, q \in \mathcal{P}$.

Remark 4.4. The similarity indicator δ_{pq} encapsulate information about the network structure into the Kriging framework. There are a number of other choices for the distance measure (e.g., the portion of overlapping sections), and link/path characteristics, such as free-flow travel time and capacity. In Chapter 5, we will present extensions on distance measure/function. Further study should compare these different measures concerning the performance of their respective Kriging predictors.

We denote i to be the time interval for which the delay needs to be estimated. Next, we define

$$\boldsymbol{d}_{p,i}(\boldsymbol{h}^1, \boldsymbol{h}^2) = \left(\delta_{pq} \left\| w^i \circ (\boldsymbol{h}_q^1 - \boldsymbol{h}_q^2) \right\|_2^2 : q \in \mathcal{P}\right) \in \mathbb{R}^{|\mathcal{P}|} \quad \forall\, p \in \mathcal{P}, 1 \le i \le n \tag{4.4}$$

where $\boldsymbol{h}_p^1 - \boldsymbol{h}_p^2 \doteq (\boldsymbol{h}_{p,i}^1 - \boldsymbol{h}_{p,i}^2 : i = 1, \ldots, n) \in \mathbb{R}^n$, and the operation $\circ$ represents component-wise multiplication. The parameter $w^i \in \mathbb{R}^n$ is defined as

$$w^i = \left(w_1^i, \ldots, w_n^i\right) \quad \text{where} \quad w_j^i = 1 \text{ if } j \le i \text{ and } w_j^i = 0 \text{ if } j > i. \tag{4.5}$$

We name $\boldsymbol{d}$ defined in (4.4) shared-link distance. This is a customized non-Euclidean distance for the Kriging metamodel we build. There are a number of extensions on distance function d to suit wilder scope of use on this metamodeling frameworks. These extensions are given in Chapter 5. Remark 4.5 provides a detailed discussion that breaks down the terms of d in (4.4) to show the meaning of our customization on distance function aspect of Kriging metamodel. For clarification, we present an example for construction of vector w. When $n = 10$, we have for example $i = 4$ and $i = 7$:

$$w^4 = (1,1,1,1,0,0,0,0,0,0)$$

$$w^7 = (1,1,1,1,1,1,1,0,0,0)$$

Remark 4.5. The similarity indicator δ_{pq} accounts for the topological configuration of any two paths in the network and, therefore, reflects the potential influences of one another using spatial information. On the contrary, w_j^i accounts for the temporal correlation between the departure rates in two distinct time intervals. The way in which we define w_j^i implies the assumption that the departure rates at a later time are not considered in defining the dissimilarity at an earlier time. Note that this assumption is not entirely realistic as the departure flow along certain path can indeed affect drivers departing at an earlier time along some other path, but here, we choose (4.5) to balance between sophistication of the model and mathematical simplicity as well as dimensionality. Building on this, the distance function $\boldsymbol{d}_{p,i}(\cdot,\cdot)$ assesses the 'distance' between two feasible path flows comprehensively in both spatial and temporal dimensions.

We apply exponential covariance function defined as follows, which is the same with all paths p, but different for each time instance i for which delay is estimated:

$$\tilde{C}_{p,i}\left(\boldsymbol{d_i}(\boldsymbol{h}^1,\boldsymbol{h}^2);\boldsymbol{\theta}^{p,i}\right) = \sigma_{p,i}^2 exp\ (-\boldsymbol{d}_{p,i}^T \cdot \boldsymbol{\theta}^{p,i}) \tag{4.6}$$

where $\boldsymbol{\theta}^{p,i} = \left(\theta_1^{p,i}, \theta_2^{p,i}, \ldots, \theta_{|\mathcal{P}|}^{p,i}\right)^T \in \mathbb{R}^{|\mathcal{P}|}$, and $\sigma_{p,i}$ is the variance of the response $\tilde{\Psi}_{p,i}$ at the sampled data points.

Training and predicting procedures

We consider a set of training data $(h^k: k = 1,\ldots,K)$ and $(\Psi^k(h): k = 1,\ldots,K)$. We will state how such a set of training data should be generated, namely the design of experiments in next section. Given this data, we first learn the parameters in the covariance function through

maximum likelihood estimation (MLE) based on the assumed Gaussian process. Let $\theta^{p,i}$ be the parameters in the covariance function, which is to be learned. The MLE of $\theta^{p,i}$ is given by:

$$\hat{\theta}^{p,i} = \underset{x}{\operatorname{argmin}} \left(K \ \log \hat{\sigma}_{p,i}^2(x) + \log \left| \sum\nolimits_{p,i}(x) \right| \right) \tag{4.7}$$

$$\hat{\sigma}_{p,i}^2(x) = \frac{1}{K} \left(\phi_{p,i} - \hat{\mu}_{p,i}(x) \cdot \mathbf{1} \right)^T \left(\sum\nolimits_{p,i}(x) \right)^{-1} \left(\phi_{p,i} - \hat{\mu}_{p,i}(x) \cdot \mathbf{1} \right) \tag{4.8}$$

$$\hat{\mu}_{p,i}(x) = \frac{\mathbf{1}^T \left(\sum_{p,i}(x) \right)^{-1} \phi_{p,i}}{\mathbf{1}^T \left(\sum_{p,i}(x) \right)^{-1} \mathbf{1}} \tag{4.9}$$

$$\sum\nolimits_{p,i}(x) = \left\{ \mathcal{C}_{p,i}(\boldsymbol{h}^j, \boldsymbol{h}^k; x) \right\}_{j,k=1,\dots,K} \in \mathbb{R}^{K \times K} \tag{4.10}$$

$$\phi_{p,i} = \left(\boldsymbol{\Psi}_{p,i}(\mathbf{h}^1), \dots, \boldsymbol{\Psi}_{p,i}(\mathbf{h}^K) \right)^T \in \mathbb{R}^K \tag{4.11}$$

and $\mathbf{1}$ is the $K \times 1$ column vector consisting of one's. The *best linear unbiased predictor* (BLUP) from ordinary Kriging yields the following estimates for a given vector $\boldsymbol{h^0}$ of path departure rates:

$$\Psi_{p,i}(h^0) \approx \hat{\mu}_{p,i}\left(\hat{\theta}^{p,i}\right) + c_{p,i}^T \cdot \left(\sum\nolimits_{p,i}(\hat{\theta}^{p,i}) \right)^{-1} \left(\phi_{p,i} - \hat{\mu}_{p,i}\left(\hat{\theta}^{p,i}\right) \cdot \mathbf{1} \right) \quad \forall (p,i) \tag{4.12}$$

where $c_{p,i} = (\mathcal{C}_{p,i}(\boldsymbol{h}^0, \boldsymbol{h}^1; \hat{\theta}^{\mathrm{p,i}}), \dots, \mathcal{C}_{p,i}(\boldsymbol{h}^0, \boldsymbol{h}^K; \hat{\theta}^{\mathrm{p,i}}))^T$, and each $\mathcal{C}_{p,i}(\boldsymbol{h}^0, \boldsymbol{h}^1; \hat{\theta}^{\mathrm{p,i}})$ is given by Equations (4.2)-(4.6). Under the assumed gaussian errors in Gaussian processes, the BLUP is equivalent to *minimum mean square error predictor* (MMSE).

Experimental design of training dataset

It is well agreed that in computer code metamodeling with high-dimensional input, the quality and performance of the metamodel built strongly depends on the DOE (point location and density) (Viana et al., 2010). Our approach is developed from the standard Latin Hypercube sampling method (Fang et al., 2005), which tries to uniformly sample the feasible region of

departure rates. Recall that the feasible region derived from flow conservation constraints of an RC choice DUE is

$$\Lambda = \left\{ h(\cdot):\ h_p(t) \geq 0,\ \sum_{p \in \mathcal{P}_{rs}} h_p(t) = R_{rs}(t) \qquad \forall t \in [t_0, t_f],\ \forall (r,s) \in \mathcal{W} \right\}$$

This feasible region is simplex-like in functional space, because the summation of departure rates $h_p(t)$ on all paths between a given OD pair (r,s) is given a priori by function $R_{rs}(t)$. The feasible region on its finite dimensional counterpart is

$$\boldsymbol{\Lambda} = \left\{ \boldsymbol{h}(\cdot):\ \boldsymbol{h}_p \geq 0,\ \Sigma_{p \in \mathcal{P}_{rs}} h_{p,i} = R_{rs}^i \quad \forall\, i = 1, \ldots, n, \quad \forall (r,s) \in \mathcal{W} \right\}$$

where $R_{rs} = \left(R_{rs}^i : i = 1, \ldots, n \right)$.

We can see that the feasible region of our finite DNL model is also simplex-like in vector space. We have for each time interval i, the summation of departure rates $h_{p,i}$ on all paths between a given OD pair (r,s) is given a priori by R_{rs}^i. This feature is inherited from its continuous counterpart (continuous time DNL of the RC DUE model).

We note that our experimental region can be scaled to a simplex, and the original Latin Hypercube sampling is designed for general experimental region, with generation scheme given for sampling on the unit cube. To obtain better sampling efficiency and quality, we transfer the design that is uniform on unit cube to the simplex-like feasible region uniformly. There has literature published on mixture design (Ning et al., 2011; Fang and Wang, 1993; Wang and Fang, 1996). For example, transfer methods are proposed for transferring a design on unit cube to design on simplex by (Fang and Wang, 1993). In our study, we have a space-filling design by which we revised the standard LHS to samples uniformly on a medium to high-dimensional simplex.

A space-filling experimental design

Certain methodology in experimental design should be applied to generate training data for our problem. The goal of experimental design is to uniformly sample the input variables from their respective domains. In this section we discuss in detail our space-filling experimental design.

The uniformity of a design or sampling strategy can be measured by many different criteria including mean square error, discrepancy, and so forth. For our specific problem, we apply a space-filling sampling strategy adapted from Latin Hypercube Design (LHD) (McKay et al., 1979; Tang, 2008; Fang et al., 2005; Santner et al., 2013) The advantage of LHS-based sampling, compared to simple random sampling or Monte Carlo sampling, is that it has a smaller variance of the sample mean and lower discrepancy, which means better uniformity in the experimental region (Fang et al., 2005; McKay et al., 1979).

Our procedure is to systematically generate the training data on a simplex induced by some flow conservation constraints, which is aligned with the RC DUE framework. Specifically, given the OD departure rate vector $R_{rs} = \left(R_{rs}^i : i = 1, \ldots, n\right)$ for each OD pair $(r, s) \in \mathcal{W}$, we recall the following constraints for path-specific departure rates h_p:

$$\Sigma_{p \in \mathcal{P}_{rs}} h_{p,i} = R_{rs}^i \quad \forall\, i = 1, \ldots, n, \quad \forall (r, s) \in \mathcal{W} \tag{4.13}$$

We see that such constraints are decoupled for different time intervals i and different OD pairs (r, s). Taking advantage of such a time-independency, we employ the Latin Hypercube Sampling method following Fang et al., (2005) and Santner et al., (2013).

We begin with the following definition of Latin Hypercube Design (LHD):

Definition 4.1. (Fang et al., 2005) An LHD design with M runs and N input variables, denoted by LHD (M, N), is a M × N matrix. In such a matrix, each column is a random permutation of $\{1, 2, \cdots, M\}$. The following algorithm, based on the LHD and (Fang et al., 2005),

summarizes the procedure employed to generate the training data uniformly within the simplex expressed in Equation (4.13).

Algorithm 1: LHD-based training data generation for OD pair *(r, s)* and time interval i

Step 1 For integers {1,…,K} where K is the prescribed number of samples, independently generate $|\mathcal{P}_{rs}|$ number of random permutations: $\{\pi_p(1), \dots, \pi_p(K)\}$ for $p \in \mathcal{P}_{rs}$. These permutations constitute an LHD (K, $|\mathcal{P}_{rs}|$).

Step 2 Generate $K \times |\mathcal{P}_{rs}|$ i.i.d. uniformly distributed variables $\{U_k^p : k = 1, \dots, K, p \in \mathcal{P}_{rs}\}$ between 0 and 1 and let

$$\lambda_k^p = \frac{\pi_p(k)}{K} - \frac{U_k^p}{K} \quad k = 1, \dots, K, \quad p \in \mathcal{P}_{rs}$$

Step 3 For each $1 \le k \le K$, the departure rates at time interval *i* are assigned as:

$$h_{p,i}^k = R_{rs}^i \frac{\lambda_{\mathrm{k}}^p}{\sum_{q=1}^{|\mathcal{P}_{rs}|} \lambda_k^q} \quad p \in \mathcal{P}$$

Adaptive sampling of the experimental domain

The LHD-based algorithm presented above tends to sample the training dataset uniformly from the simplex structure (4.13). However, when the response surface is sufficiently irregular such uniform sampling strategy may not be ideal. To further improve the performance of the training phase, we apply an adaptive sampling procedure to iteratively update the training set until a target error rate is achieved.

This will result in a better exploration of the experimental domain especially in areas where the response surface has low regularity. The procedure is as follows.

Algorithm 2 Iterative update of the training set

Step 1 Fix integers 0<M<K, $\delta > 0$, and error threshold $\epsilon > 0$. Generate an initial training set $\mathcal{T}_0$ of size K based on Algorithm 1. Select a target error threshold ε. Set the iteration counter $j = 0$.

Step 2 Train the model using training set $\mathcal{T}_j$. Generate a uniformly sampled testing dataset S_j of size M using Algorithm 1.

Step 3 Test the trained model using the testing dataset S_j. Find the subset $\mathcal{E}_j \subset S_j$ that contains testing data points with errors above ε.

Step 4 If $|\mathcal{E}_j| \leq \delta$, stop and output $\mathcal{T}_j$. Otherwise, update training set as $\mathcal{T}_{j+1} = \mathcal{T}_j \cup \mathcal{E}_j$. Set $j = j + 1$ and go to Step 2.

Improvement in experimental design and sampling

An effective experimental design, in our problem, is significant in the following aspects:

- results in better model fit quality and lower prediction error
- reduces the number of training data required for a satisfying model performance
- reduces computational effort in training data generation effort and model training time

Also, we have been discussing experimental design/training set generation approaches for general use of the learned DNL metamodel. For a specialized use of DNL metamodel, e.g., for faster DUE computations, it is possible that more information can be obtained for the region of interest within the entire design region. Instead of trying to fit the entire region, sequential sampling approaches can be applied to explore the experimental region in an adaptive way (Chen et al., 2014; Jones et al., 1998). This will significantly improve the efficiency of the model fitting in both time and memory consumption, especially when working with large-scale problems. In summary, our effort to improve the sampling process are to employ state-of-art algorithms based

on Latin Hypercube sampling, sampling on simplex and mixture design, and sequential experiments.

Numerical studies

In this section, the goal is to numerically evaluate the performance of the surrogate DNL models obtained from the proposed metamodeling with Kriging. We are evaluating approximation accuracy and computational efficiency. For the conventional and exact DNL model used in generating both the training and testing datasets, we employ the discretized Lighthill–Whitham–Richards network model equivalent to the link transmission model (Yperman et al., 2005). The detailed DNL procedure based on this model is presented in Han et al. (2016a). The link transmission model is among the most efficient computational algorithms for propagating flow and congestion on a network level, and captures realistic phenomena such as physical queues and spillback. For the training phase of Kriging, we adapt the well-known DACE Matlab toolbox (Lophaven et al., 2002) by incorporating the customized distance and correlation functions defined in Equations (4.4) and (4.6). The MLE (Equation (4.7)) is performed with an iterative pattern search algorithm. Moreover, the generation of the training data follows algorithms 1 and 2.

Test scenarios

We test the performance of the Kriging approach based on five network scenarios: (1) The 19-arc network with 4 OD pairs and 24 paths, which is studied in Nguyen (1984) and Nie and Zhang (2010) and hereafter referred to as the Nguyen network; and (2) The Sioux Falls network (Friesz et al., 2011; Suwansirikul et al., 1987) with 60, 119, 201, and 501 paths,

respectively. The Nguyen network and the Sioux Falls network are shown in Fig. 1-1 and Fig. 1-2. The time horizon of all DNL models is [0, 5] (in hours). Two values n = 5 and n = 10 are considered where n is the number of time intervals used in the construction of the finite dimensional delay operator.

To evaluate the accuracy of the surrogate delay operators obtained from Kriging, testing data sets that contains path departure rates not used to fit the model are calculated and collected for each surrogate delay operator. Specifically, for each network scenario, 100 testing data points are randomly generated over the entire feasible region. To separate testing data from the training data to get the most reliable testing result, all testing data points are newly and separately generated in the testing phase and no training data is re-used in testing.

For the testing dataset, we compare outputs (i.e., predicted path travel times) from the surrogate model with those from the exact delay operator Φ. We denote one testing data as h_0. The prediction error corresponding to one testing data h_0 is calculated as the prediction error, and speed-up index for one instance of prediction is defined as the ratio between the computational time of the exact delay operator Φ and the prediction time of Φ (not including training time). These two criteria are defined as follows:

$$Prediction\ Error\ = \frac{\|\Phi(h_0)-\Psi(h_0)\|_2}{\|\Psi(h_0)\ \|_2} \times 100\%.$$

$$Speed-up\ index = \frac{\text{computational time of the exact delay operator } \boldsymbol{\Psi}}{\text{prediction time of } \widetilde{\boldsymbol{\Psi}}}$$

Over the newly generated 100 testing data, the prediction errors and speed-up indices are averaged for presentation in Table 4-1. It is to be noted that the prediction time of $\widetilde{\boldsymbol{\Psi}}$ does not include training time. The larger the speed-up index, the more computationally efficient our surrogate model is compared to the conventional DNL.

Test results for the Kriging-based delay operator

The performance of the Kriging-based delay operators Φ is summarized in Table 4-1 where, for each test scenario, a smaller and a larger dataset are used to train Φ. It can be seen that the accuracy of the approximate DNL models increases at some expense of decreased efficiency when the training dataset becomes larger. Such a trend is further illustrated in Fig. 4-3 for the Nguyen network and Fig. 4-5 for the Sioux Falls network (with 119 paths), where results based on a wider range of training dataset sizes are available. It is also clear from Table 4-1 and Fig. 4-4 and Fig. 4-6 that the surrogate DNL models yield accurate prediction of the path travel times, with errors typically below 8% for the Nguyen network and, interestingly, below 3.4% for the Sioux Falls networks. The lower errors for larger networks are likely to be caused by the selection of training data; that is, algorithms 1 and 2 sample the design domain more exhaustively for the Sioux Falls networks than the Nguyen network. The excellent prediction errors is reasonably explained by the adaptive sampling and the space filling generation of training set we particularly designed for the DNL. Based on the structure of the original model and characteristic of Kriging, another possible contributing factor to the low error is the flow conservation constraints of the DNL, which makes the problem a constrained one. Because all dynamic traffic assignment models assume a constrained feasible region, this contributing factor is certainly to be retained and effective should other types of the DNL model is learned. Also, one can adaptively reduce the training datasets for the Sioux Falls networks to gain more computational efficiency in the training phase without significantly increasing the errors. This highlights the robustness of the Kriging method with flexible trade-offs between model accuracy and complexity.

The surrogate DNL models yield significant computational savings compared to the exact DNL models, with speed-ups ranging from 9 to 455 (times faster). This far superior computational efficiency is achieved at only very minor prediction error, showing a very effective

tradeoff between model accuracy and complexity. In addition, the results for the Sioux Falls network with 501 paths shows that the proposed metamodeling methodology is capable of handling large-scale and high-dimensional problems.

In Fig. 4-4 and Fig. 4-6, for some example paths, we show the exact and predicted path delays on the Nguyen network and the Sioux Falls network. It is clearly seen that the surrogate DNL models approximate the exact ones quite well, with errors similar to those shown in Table 4-1. The situations on the other paths are similar and are not shown exhaustively here.

$n = 5$	Nguyen network (24 paths)		Sioux Falls (60 paths)		Sioux falls (119 paths)		Sioux falls (201 paths)		Sioux falls (501 paths)	
# of training data	100	200	100	200	100	200	100	300	100	300
Training time	30 s	121 s	270 s	1053 s	30 min	151 min	99 min	873 min	32 h	221 h
Prediction error	6.7%	5.3%	2.5%	1.9%	3.3%	2.7%	1.9%	1.7%	3.2%	2.8%
Speed-up index	181	89	143	77	455	159	21	21	91	31
$n = 10$	Nguyen network (24 paths)		Sioux falls (60 paths)		Sioux falls (119 paths)		Sioux falls (201 paths)		Sioux falls (501 paths)	
# of training data	120	240	120	240	120	240	150	200	200	300
Training time	91 s	351 s	777 s	3116 s	105 min	386 min	458 min	752 min	449 h	497 h
Prediction error	7.1%	5.6%	2.9%	2.2%	3.3%	2.9%	2.2%	2.2%	3.3%	2.9%
Speed-up index	38	20	42	19	132	66	12	9	13	9

Table **4-1** Performances of the surrogate delay operator Φ: the prediction errors and speed-up indices are averaged over 100 randomly sampled testing data

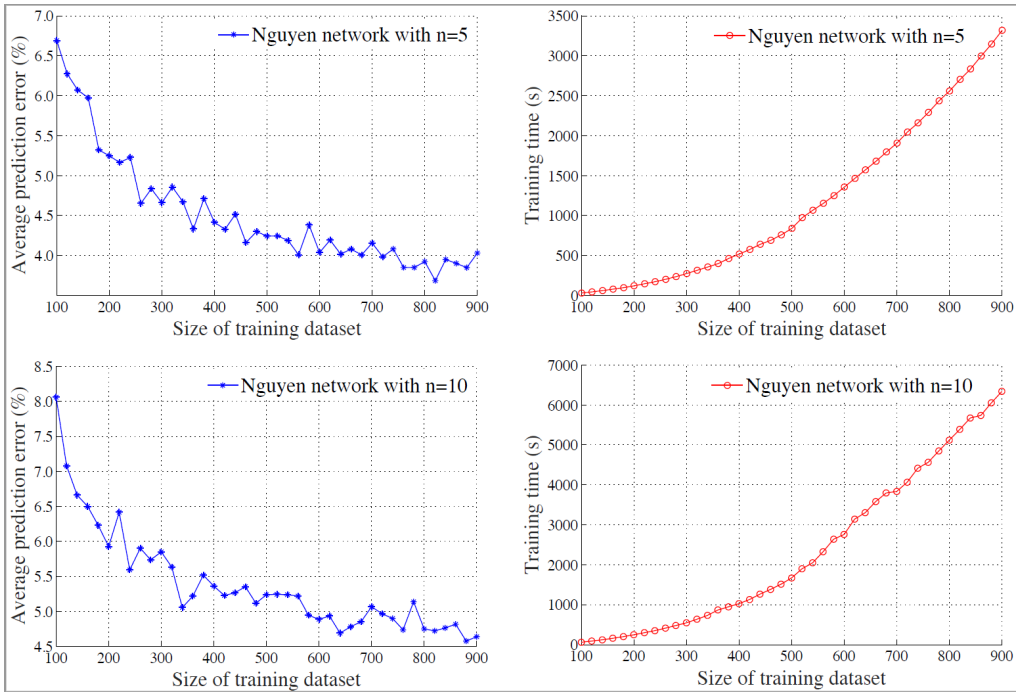

Figure **4-3** Prediction trend of Nguyen network: prediction errors and training time as size of training dataset increases

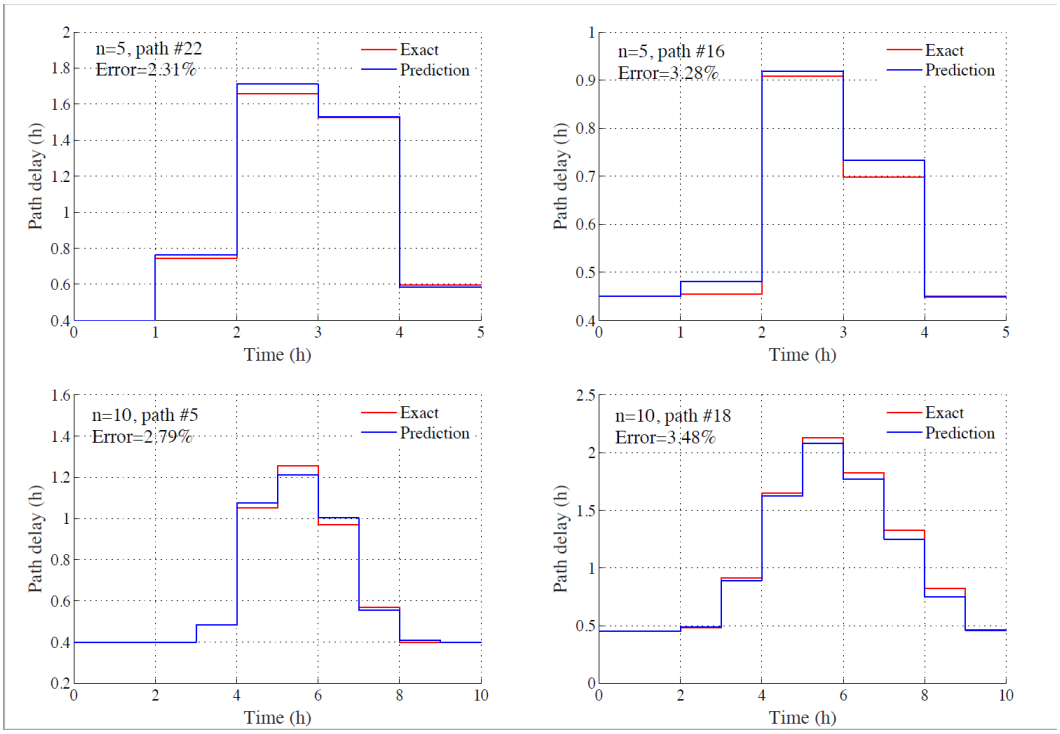

Figure **4-4** The Nguyen network: comparison of exact and predicted path delays on example paths.

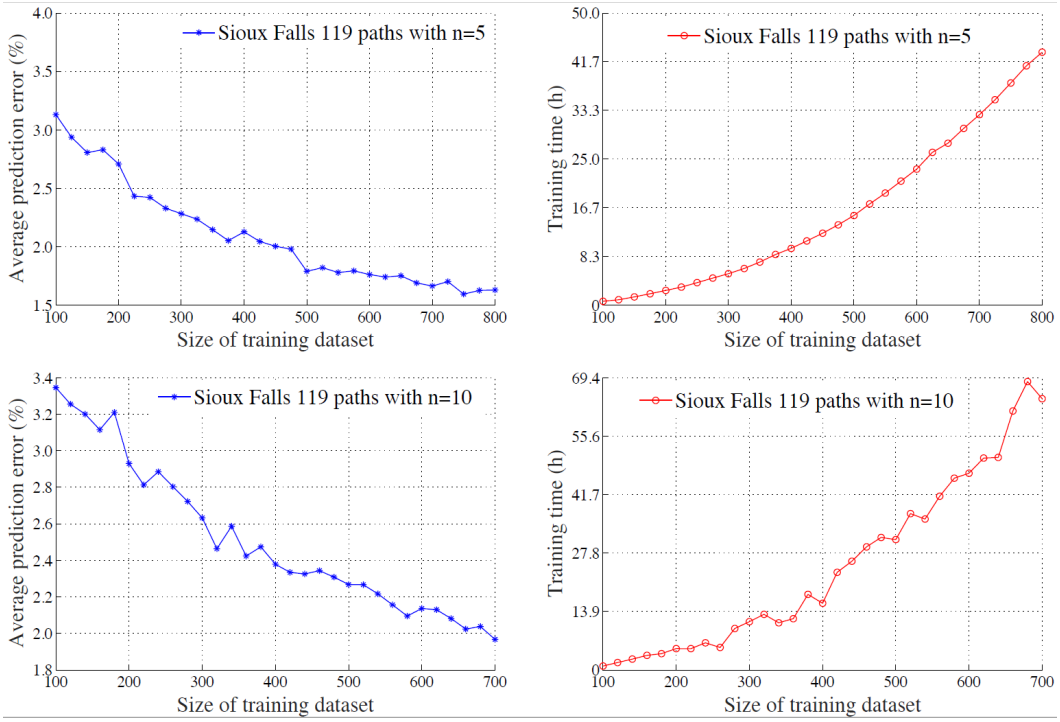

Figure **4-5** Prediction trend of Sioux Falls network: prediction errors and training time as size of training dataset increases

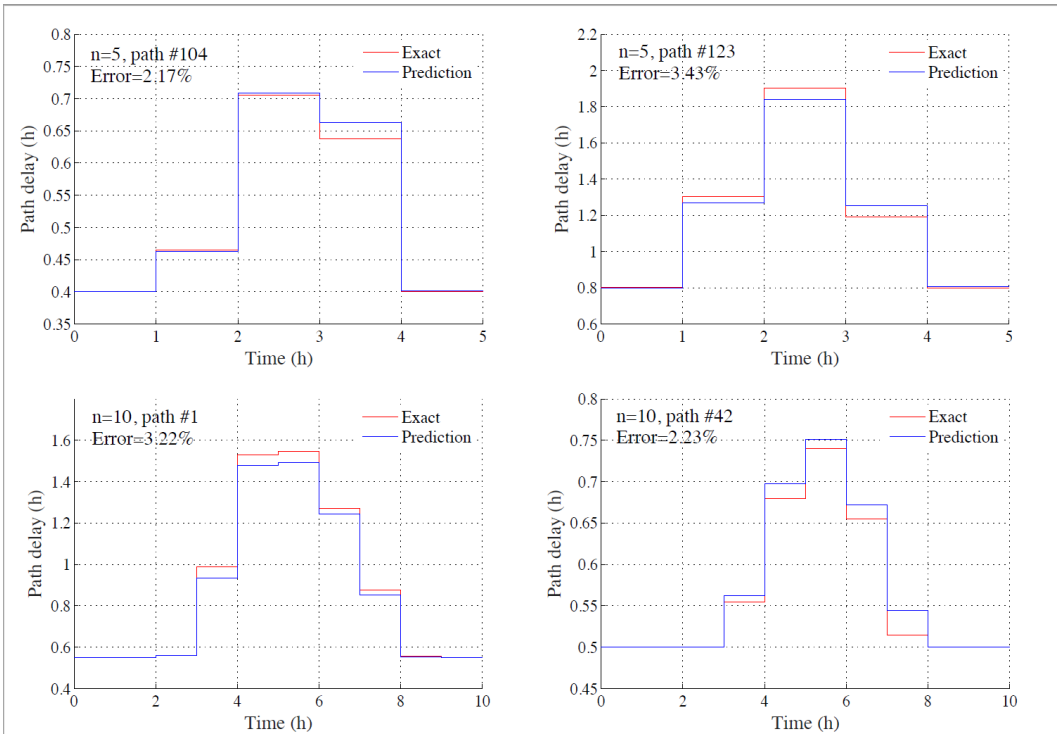

Figure **4-6** Sioux Falls network with 501 paths: comparison of exact and predicted path delays on example paths

Comparison to Neural Networks

A comparison of metamodeling performances between Kriging and neural networks (NN) is conducted. The test scenario involves the Nguyen network and n = 5. Both Kriging and the NNs are fitted using the same set of training datasets, with sizes ranging from 100 to 880. Afterwards, their prediction performances are compared using the same test data consisting of 100 randomly generated samples. The NN has two layers and is trained with the Levenberg–Marquardt algorithm. We use Matlab's R2016 Neural Networks toolbox for this task. Fig. 4-7 (left) shows the average prediction error provided by each method on the testing dataset. The bars denote two standard deviations from the prediction means. The overlap of the prediction intervals

indicates that differences in prediction error between the two methods are not statistically significant. Fig. 4-7 (right) shows the training times to build the surrogate models. Again, no significant different is found between the computational efforts required as a function of the size of the training dataset. Based on the preliminary comparison on this test case, we can see that for about the same computational training time, Kriging and NN metamodels perform similarly. As mentioned before, Kriging has two main advantages which justify its use: it provides closed-form, analytic expressions for the predictions and it is an exact interpolator, which means that predictions at the trained data points are not smoothed but are identical to the observed (i.e., the prediction error for the training dataset is zero). These are not properties shared by NNs.

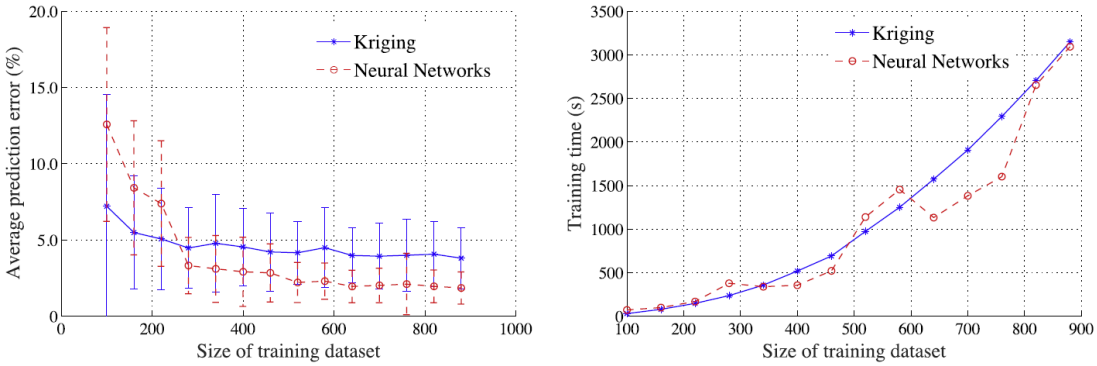

Figure **4-7** Comparison of Kriging and NN on the Nguyen network: the vertical bars denote two standard deviations from the prediction means

Mathematical properties

The proposed metamodeling framework is applicable to a wide range of DNL models with a variety of link dynamics, junction models, and path delay models. It underpins a new generation of network performance models that can serve as efficient and tractable alternatives to the otherwise exact models, and their reliability can be improved with in-depth analysis of the covariance functions, parameters, and methods that generate training data (Li and Sudjianto,

2005). The computational efficiency of the DNL models resulting from the metamodeling approach based on Kriging, which is demonstrated in the numerical studies section, could significantly speed up DTA computations for large-scale networks. Moreover, the corresponding delay operators have closed-form expressions, which is potentially beneficial for DTA modeling, and this section will explore these opportunities. Throughout this section, we use Φ: $\mathbb{R}^{n\times|\mathcal{P}|} \rightarrow \mathbb{R}^{n\times|\mathcal{P}|}$: to denote the surrogate delay operator obtained from Kriging.

Closed-form expression of the delay operator Φ

Given a training dataset consisting of K samples, the predictor, expressed in Equation (4.12), can be written in a concise form as:

$$\boldsymbol{\Phi}_{p,i}(\boldsymbol{h^0}) = X^{p,i} + c_{p,i}(\boldsymbol{h}^0)^T \cdot Y^{p,i} \tag{4.14}$$

where h_0 is the input path flow vector. Both $X^{p,i} \in \mathbb{R}$ and $Y^{p,i} = (Y_k^{p,i}: k = 1\ldots, K) \in \mathbb{R}^K$ only depend on the training data and thus can be treated as constant once the training phase is finished. Moreover, the expression for $c_{p,i}$ $(\boldsymbol{h}^0)$ is given by Equations (4.2)-(4.6):

$$c_{p,i}(\boldsymbol{h}^0) = \sigma_{p,i}^2 \left(exp\left(-\boldsymbol{d}_{p,i}(\boldsymbol{h}^0, \boldsymbol{h}^1)^T \cdot \boldsymbol{\theta}^{p,i}\right), \ldots, exp\left(-\boldsymbol{d}_{p,i}(\boldsymbol{h}^0, \boldsymbol{h}^K)^T \cdot \boldsymbol{\theta}^{p,i}\right)\right) \tag{4.15}$$

$$c_{p,i}(\boldsymbol{h}^0) \in \mathbb{R}^K.$$

Recall that

$$\boldsymbol{d}_{p,i}(\boldsymbol{h}^0, \boldsymbol{h}^k) = \left(\delta_{pq} \| w^i \circ (\boldsymbol{h}_q^0 - \boldsymbol{h}_q^k) \|_2^2 : q \in \mathcal{P}\right) \in \mathbb{R}^{|\mathcal{P}|} \qquad k = 1, \ldots, K \tag{4.16}$$

and $w^i \in \mathbb{R}^n$ is such that $w^i = (w_1^i, \ldots, w_n^i)$ where $w_j^i = 1$ if $j \leq i$ and $w_j^i = 0\ if\ j > i$.Based on (4.14)-(4.16), it is easy to derive the following closed-form expression:

$$\boldsymbol{\Phi}_{p,i}(\boldsymbol{h^0}) = X^{p,i} + \sigma_{p,i}^2 \sum_{k=1}^{K} Y_k^{p,i} exp\left(-\sum_{q\in\mathcal{P}} \delta_{pq}\theta_q^{p,i} \sum_{j=1}^{n} w_j^i \left(\boldsymbol{h}_{q,j}^0 - \boldsymbol{h}_{q,j}^k\right)^2\right) \quad (4.17)$$

$\forall\, p \in \mathcal{P}, i = 1, \ldots, n.$

Here, $\boldsymbol{h}_{q,j}^0{}'$s are input variables of the delay operator; $X_{p,i}$, $\sigma_{p,i}$, $Y_k^{p,i}{}'s$, $\theta_q^{p,i}{}'s$ and $\boldsymbol{h}_{q,j}^k{}'s$ are all treated as constants in the prediction of path delays.

Analytical properties of the surrogate delay operator Φ

Continuity and smoothness

The delay operator expressed in Equation (4.17) is continuous and smooth (i.e., infinitely differentiable) because it involves only elementary summation, multiplication, and exponential operations. Unlike conventional DNL models for which the discontinuity may fail when spillback occurs (we refer the reader to Szeto (2003) and Han et al. (2016a) for some examples), the continuity and smoothness of the proposed surrogate delay operator Φ will always hold regardless of the underlying link, junction, or network flow dynamics. The continuity of Φ can be used to prove the existence of solutions to DTA problems when the DNL problem is replaced with the approximate surrogate model we have presented so far.

Differentiability and Lipchitz continuity

We can also easily differentiate the delay operator and obtain its Jacobian matrix as follows. Recall that the surrogate delay operator is a mapping from the n ×|$\mathcal{P}$| -dimensional Euclidean space into itself:

$$\boldsymbol{\Phi}(\boldsymbol{h}^0) = \left(\boldsymbol{\Phi}_{p,i}(\boldsymbol{h}^0):\ \forall (p,i)\right)$$

Therefore, individual entries of the Jacobian matrix can be calculated as:

$$\frac{\partial \boldsymbol{\Phi}_{\mathrm{p,i}}}{\partial(\boldsymbol{h}_q^0, j)} = -2\sigma_{p,i}^2 \delta_{pq} \theta_q^{p,i} w_j^i \sum_{k=1}^{K} Y_k^{p,i} \exp\left(-\sum_{\hat{q}\in\mathcal{P}} \delta_{p\hat{q}}\, \theta_{\hat{q}}^{p,i} \sum_{\hat{\jmath}=1}^{n} w_{\hat{\jmath}}^i \left(h_{\hat{q},\hat{\jmath}}^0 - h_{\hat{q},\hat{\jmath}}^k\right)^2 \right) \left(h_{q,j}^0 - h_{q,j}^0\right)$$

(4.18)

Due to the finite dimensional nature of this problem, it can be easily shown that the partial derivatives (4.18) are uniformly bounded. Therefore, the delay operator Φ is in fact Lipschitz continuous. The differentiability and closed-form Jacobian of the delay operator Φ will have a number of important applications including, but not limited to, the following.

• All gradient-based methods can directly benefit from the explicit Jacobian of the delay operator. These include single- level problems (e.g., DUE, DSO), and bi-level problems (e.g., dynamic mathematical program with equilibrium constraints) that can be reduced to single-level problems (e.g., via the KKT conditions). All these problems can be directly solved by commercial solvers such as GAMS (e.g., see Friesz et al., 2007).

• The explicit differentiation of the delay operator also facilitates sensitivity analysis of dynamic network traffic equilibria (Chung et al., 2014), and benefits network design heuristics based on it. For example, the heuristic network design based on sensitivity analysis from Suwansirikul et al. (1987) can be adapted to treat dynamic traffic networks.

Generalized monotonicity

Generalized monotonicity, such as monotonicity, strong monotonicity, pseudo-monotonicity, quasi-monotonicity, and dual solvability, is necessary for the convergence of computational algorithms for DUE problems (see Han et al., 2015 for an overview of these notions). However, conventional DNL models do not allow insights regarding generalized monotonicity for general networks due to the lack of analytical representations of the delay

operators. With the closed-form expression of the delay operator Φ, it is possible to conduct rigorous analysis regarding monotonicity. We take (strong) monotonicity as an example. An operator Φ is said to be (strongly) monotone if

$$\langle \Phi(\mathbf{h}^1) - \Phi(\mathbf{h}^2), \quad \boldsymbol{h}^1 - \boldsymbol{h}^2 \rangle \geq 0 \quad (\geq \lambda \|\boldsymbol{h}^1 - \boldsymbol{h}^2\|^2 \; for \; some \; \lambda > 0) \;\; \forall \boldsymbol{h}^1, \boldsymbol{h}^2 \in \Lambda \tag{4.19}$$

Although it is not likely that the operator Φ satisfies these conditions in its entire domain, its closed-form expression enables us to identify regions where it is strongly monotone. For instance, thanks to the closed-form representation of Φ the left-hand side of Equation (4.19) can be also expressed in closed form, making it sufficient to solve an algebraic equation to identify the strongly monotone region of Φ. In addition, due to the closed-form expression and differentiability, methods based on first-order expansion of Φ can be similarly applied, enabling drawing insights into its generalized monotonicity. The closed-form expressions of Φ and its Jacobian allow us to inspect, in a quantitative way, a wide spectrum of convergence conditions proposed in the literature (Han and Lo, 2002; Konnov, 2003; Long et al., 2013), and to devise tailored numerical schemes that take advantage of such information, which is unavailable through conventional DNL models. This approach holds much promise in tackling the convergence challenges that have stymied researchers for decades, and will be pursued in a future study as it is out of the scope of this dissertation.

Chapter 5

Extensions: Distance metrics, covariances, and spatio-temporal Kriging

This chapter provides a discussion on various distance metrics and covariance functions in Kriging, including their similarities, differences, pros, and cons in the practical modeling of real-world traffic problem. By providing alternatives, we expand the practical use of the Kriging model into wider range of dynamic transportation models.

Distance metrics

Distance metric plays very important roles in the proper functioning of the statistical metamodel. A well-defined distance metric also provides sufficient flexibility for addressing complex problem with realistic factors of variety of categories and data structure to be considered. Euclidean distance is the widest used distance, but metrics may not be limited to be Euclidean. For many statistical learning objects, observations are related in feature that are not defined on Euclidean space. Non-Euclidean distance are usually proposed for the purpose of customize the distance measuring method to better reflect these features in the surrogate model. For Kriging models, various non-Euclidean distance metrics are applied in many fields such as water distance (Rathbun 1998, Lønd and Høst 2003), geodesic distance (Tenenbaum et al. 2000, Banerjee 2005), stream network distance (Peterson et al. 2007), landscape distance (Lyon et al. 2010), and wireless sensor network distance (Umer et al. 2010).

The following table list selected articles that use or discuss non-Euclidean distance metric.

Literature	Model	Distance metric	Covariance Function

Zou et al. 2011	local universal Kriging	approximate road network distance (ARND)	Exponential/classic
Rathbun 1998	Universal Kriging	Euclidean distance and water distance	Exponential and spherical
Lønd and Høst 2003	spatial random field	water distance	Gaussian
Banerjee 2005	spatial regression	geodetic distance	Exponential and Matérn
Peterson et al. 2007	N/A	hydrologic distance (stream network distance)	moving average autocovariance and exponential covariance
Lyon et al. 2010	Ordinary Kriging and Kriging with external drift	landscape distance	exponential
Umer et al. 2010	Kriging	wireless sensor network distance	spherical, Gaussian and exponential

Table **5-1** Studies that discuss or apply non-Euclidean distance metric

There are two ways to move forward from development of a customized distance metric toward a full model with covariance function. Some studies use isometric embedding in mathematics, which means only to use transformation of the spatial coordinates, and adopting the classical spatial covariance function (e.g., Zou et al., 2012); Another approach is to apply kernel convolution or the moving average function, and deduce a new "customized" covariance function.

We propose in this chapter a generalized form of distance metric that extent the scope of the model developed in Chapter 4, providing increased flexibility, broadened horizon of usage and limit-expanding solutions to computationally expensive models that need special skills in addressing beyond scope of the standard version.

The similarity between two paths is defined as

$$\delta_{pq} \in \mathbb{R} \quad \forall\, p, q \in \mathcal{P}$$

$$\mathcal{P}\ is\ the\ set\ of\ paths$$

Based on the path similarities, we recall the distance metric defined in previous chapter as

$$\boldsymbol{d}_{p,i} = \left(\delta_{pq} \left\| w^i \circ (\boldsymbol{h}_q^1 \quad - \boldsymbol{h}_q^2) \right\|_2^2 : q \in \mathcal{P}\right) \in \mathbb{R}^{|\mathcal{P}|} \qquad \forall\, p \in \mathcal{P}, 1 \leq i \leq n$$

Regular similarity for Kriging

$$\delta_{pq} = \rho_{pq} = \frac{\#\ of\ shared\ links\ between\ path\ p\ and\ path\ q}{average\ \#\ of\ share\ links\ between\ path\ p\ and\ all\ paths\ q' \in \mathcal{P}}$$

We call ρ_{pq} Shared-Link Similarity (SLS).

The regular similarity for network paths account for similarity induce by sharing links and segments (consecutive links) in common. The idea is intuitive because paths that shares common segments will share flows at any time, any location on the segment. Therefore, a natural layer of correlation between delays on both paths is underlying following the identified similarity between construction of both paths. We also note that for a complex network, paths that does not share links directly can also be correlated in delays through a media path that share links with both, or through passing through a common area of interacting traffic. These “second degree” similarity can be accounted by more general form of similarity introduced in following sections.

Revised similarity for Kriging

$$\delta_{pq} = \lambda\rho_{pq} + (1-\lambda)\tau_{pq}$$

Where $\rho_{pq} = SLS$ of p and q.

$$\tau_{pq} = absolute\ value\ of\ differences\ between\ delays\ on\ path\ p\ and\ q$$

We define τ_{pq} as Delay Similarity (DS)

Note that $\tau_{pq} \to 0$ as the system goes to UE. Therefore, we have

$\delta \to \delta_{regular}$ as the system goes to UE.

A natural extension of τ is the Effective Delay Similarity (EDS), defined as $\tilde{\tau}$

$$\tilde{\tau} = \tau + absolute\ value\ of\ differences\ between\ \mathcal{F}\ on\ path\ p\ and\ path\ q$$

Note that both DS and EDS is not static over the network, but are dynamic variables that defined on time and the delay Ψ. This will add one more layer of complexity to the model in computation efficiency. To have the Kriging method trainable offline, we recommend an approximation or prediction to the DS and EDS to be applied, rather than regenerated each iteration.

DS and EDS are designed for DUE models and bi-level models that have an equilibrium lower level (e.g. an MPEC). Modification in adding a term that represent similarity in user's payoffs reflect the Nash-like game nature of traffic equilibrium models. In a DUE computation, only evaluations of dynamic network loading (DNL) that is along the path (iterations) toward equilibrium is of interest. In the entire feasible region defined in for departure rates, training more intensely in areas that is closer to the equilibrium than in areas far way will be beneficial. Comparing to training blindly in all areas, avoid areas that is never visited during the DUE computation is more strategic and efficient. DS and EDS are new distance metrics that can benefit from the convergence of DUE computation, during which the absolute value of differences between any pair of paths reduces to zero as the systems converges to equilibrium.

SLS similarity with higher degree

We noticed that the SLS introduced in the above section accounted for two paths that directly share links. We now extend the concept to higher degree of link similarity that account for two paths that does not share links directly, but are correlated through intermediate paths.

Definition 5.1 SLS with degree is similarity between two paths on a network. Define a SLS degree as the minimum number of paths in between from one path to another, where each pair of paths has at least one shared link. This means that any two paths with a SLS of degree k will have paths with SLS of $k-1, k-2, \ldots 1$ in between. Denote the degree of SLS using Roman Numerals. For example, an SLS of degree 6 is denoted by ρ_{pq}^{VI}.

SLS with degree describe path correlations with better hierarchy. It distinguishes levels of similarities between different degrees of link-sharing or path overlapping. It reflects the characteristic of paths correlations on a traffic network "flow on all paths are related to each other, but the paths that are connected through less intermediates are likely to be more related".

Affine similarity metric

This metric works for model with m attributes. It is defined as follows

$$\delta_{pq} = \sum c_i a_i$$

$$\sum c_i = 1$$

$$i = 1 \ldots m$$

where δ_{pq} is a linear and convex combination with m criteria terms that describes a weighted sum of differences between path p and path q. We name this similarity Affine Similarity (AS).

AS can take a special form of

$$\delta_{pq} = \lambda \rho_{pq} + (1 - \mu - \lambda)\tau_{pq} + \mu a_{pq}$$

That includes both ρ and τ when $a_1 = \rho$ and $a_2 = \tau$.

AS extends path similarity into a form that can include any number of attributes that reflect the shared criteria between two paths. These criteria can be topological or non-topological. This form presents the most flexible framework that one can include any specific criteria that needs taken into account. The criteria can be topological, environmental, or network economical. The criteria will be predefined when the model is built. Examples of terms in the metric are degree of closeness, differences in tolls, road capacity, speed limits, number of shared intersections, road preferences, and possibility of congestion. A reasonable inclusion of the affecting criteria will properly address the important effects that have an influence on the modeling, and speed up the learning process with better structure of surrogate model.

As discussed in chapter 4, when we replace distance metric d with customized distance metric in ordinary Kriging. Metamodel with customized distance with have a different form of BLUP of the parameter θ. Mathematical derivation is not included in this dissertation.

Benefit of new distance metrics

It is worth investigating the effect of distance metric replacement on metamodel. In our approach, upgraded distance metrics with alternative similarities do not affect the validity of metamodel. The prediction accuracy Kriging model enjoys is expected to be furtherly improved on customized similarity function.

The speed of solving solution relies on the starting point of θ and mathematical complexity of the model. Because all similarities we proposed are linear, the superior speed we observed is expected to be maintained. Additional terms added may affect the computational speed in modeling-training. Model training can be done offline. This disadvantage is also offset

by the benefits a well-customized distance function brings. The prediction accuracy Kriging model enjoys will not be decreased nor sensitive to the choice of a similarity function.

Validity

Validity of a distance determines the scope of its proper use. The proper perform of the Kriging method depends on careful choice of distance metric. Distance metric $d(x_1, x_2)$ defined in (4.4) is a valid metric measure or a valid metric when it satisfies (1)-(3) among the following requirements, and is a translation invariant metric when it satisfy (4). A distance metric is a valid pseudo-metric under relaxed conditions from the metric requirements. A detailed proof is provided in the appendix for validity.

1. $d(x_1, x_2) = d(x_2, x_1)$
2. $d(x_1, x_2) \geq 0 \; and \; d(x_1, x_2) = 0 \; iff \; x_1 = x_2$
3. $d(x_1, x_2) \leq d(x_1, x_3) + d(x_3, x_2)$
4. $d(x_1 + \alpha, x_2 + \alpha) = d(x_1, x_2)$

$$\forall \; x1, x2 \; in \; domain \; of \; the \; metric$$

In summary of the proof, the distance metric defined in Equation (4.4) satisfies pseudo-metric requirements and enjoys mathematical properties of a pseudo-metric. For certain networks, it satisfies all requirements and is a valid distance metric mathematically. The detailed proof and discussion on validity of our proposed distance metric based on proposed similarity between paths is given in appendix A.

In addition, we need to have a valid covariance function using the distance metric defined. A valid covariance function is a positive definite function. This means a covariance function is valid when it satisfies

$$\sum_{j,k=1}^{n} a_j a_k \mathcal{C}(h^j, h^k) > 0 \ \forall a_j, a_k \in \mathbb{R} \ \forall h^j, h^k \in \Lambda$$

A proof of positive definiteness of covariance function defined on our proposed distance metrics is given in appendix B.

Remark 5.1: It is to be noted that the distance discussed here is a distance metric in the space of input variable Λ. In our metamodel, the input variable is $h \in \mathbb{R}^{n\times|\mathcal{P}|}$. Therefore, the distance metric d should satisfy symmetric requirement on vector space $\mathbb{R}^{n\times|\mathcal{P}|}$. We require that the distance metric is symmetric on any pair of input h^1 and h^2. This symmetric property of the distance metric is not defined based on the space of all paths. Hence, it should not be confused with being symmetric on any pair of paths p and q, which is not true here.

Transportation network is usually defined as a directed graph (Miller and Shaw 2001), which is used to reflect driving direction of streets. It is worth noting for clarity that directed road network distance is asymmetric and distinctly violates (1) and is therefore not a valid distance metric. Distance metric in our proposed method is a different concept. The experiment we constructed is essentially a computer experiment that is conducted on a vector space. When discussing the validity of distance, we need to emphasize about the space we are discussing upon. Remark 5.1 addresses symmetric issue of distance metric in terms of distinguishing between set of paths and the space on which the distance of our model defines on.

Covariance functions

In Kriging, covariance function is a nonincreasing function, which reflects the characteristic of spatial data: ‘everything is related to everything else, but things near are more related than things distant’ (Tobler 1970). The relationship between data samples in a Kriging model is reflected and described by the covariance functions.

In the model presented in chapter 4, the covariance between two input, h^1 and h^2 in the ordinary Kriging for all (p, i) is defined as

$$\mathcal{C}(d(h^1, h^2), \theta) = \sigma^2 \exp\left(-d(h^1, h^2)^T \cdot \theta\right)$$

After simplifying in notations, we neglect the subscript and superscript (p, i) in our subsequent discussion. The result works for all (p, i). We denote here

$$\mathcal{C}(d, \theta) = \sigma^2 \exp(-d^T \cdot \theta)$$

where d is defined in (4.4) and

$$\theta \in \mathbb{R}^{|\mathcal{P}|}$$

$$d \in \mathbb{R}^{|\mathcal{P}|}$$

$$\mathcal{C} \in \mathbb{R}$$

In our proposed metamodeling in Chapter 4, covariance takes an exponential form in ordinary Kriging. Covariance functions are required to satisfy the following property: the covariance matrix needs to be positive definite by inputting the covariance. The exponential family is commonly used as kernel functions.

We provide some alternatives in the following sections.

- Kernel function

In ordinary Kriging, we use an exponential function as the kernel function, but in realistic scenarios we may need alternative forms of kernel functions to suit the need of the target model. To deal with this challenge, we can change the kernel function in the Kriging settings.
In this session, we discuss alternative kernel functions to apply on the Kriging frameworks, and present the output model in both exponential and non-exponential forms. In the following, two classes (the polynomial kernel and triangular kernel) are discussed.

1. Polynomial kernel

$$cov(d) = \sigma^2 * \theta_a (1 - d)^{-\alpha}$$

2. Triangular kernel

Triangular kernel is a special form of polynomial kernel. The kernel function takes the following form:

$$cov(d) = \begin{cases} \sigma^2 * \theta(1-d), & d \in [0, \frac{\sigma^2}{\theta}] \\ 0, & d \in [\frac{\sigma^2}{\theta}, \infty] \end{cases}$$

Spatio-temporal Kriging

We recall the spatio-temporal random process $Y(s;\ t), s \in\ D_s, t \in\ D_t$ that evolves through the spatio-temporal index set $D_s\ \times\ D_t$ following (Cressie and Wikle, 2011):

$$Y(s;t)\ = \mu(s;t) + \beta(s) + \gamma(t) + \kappa(s;t) + \delta(s;t), \quad s \in D_s, t \in D_t \tag{5.1}$$

Because our DNL problem is deterministic, we propose to drop the white noise term in (5.1) and apply the ordinary Kriging framework (assuming a deterministic mean μ). We assume that the delay operator can be approximated by a realization of the spatio-temporal random process as follows.

$$\tilde{\Psi}_p(\boldsymbol{h};\ t) = \mu_p + \beta_p(\boldsymbol{h}) + \gamma_p(t) + \kappa_p(\boldsymbol{h};t)\ \ t \in\ [t_0, t_f], \quad \boldsymbol{h} \in \mathbb{R}_+^{n\times|\mathcal{P}|}$$

for each path $p \in \mathcal{P}$. In the model, $\beta_p(\boldsymbol{h})$, $\gamma_p(t)$, and $\kappa_p(\boldsymbol{h};t)$ have zero mean and the following covariance functions respectively for all p.

$$cov(\beta_p(h^1), \beta_p(h^2)) = \ C_p^{\beta}(h^1, h^2; \theta^{\beta,p})\ = \ \tilde{C}_p^{\beta}(d(h^1, h^2); \theta^{\beta,p})$$

$$cov\left(\gamma_p(t^1), \gamma_p(t^2)\right) = \ C_p^{\gamma}(t^1, t^2; \theta^{\gamma,p}) = \ \tilde{C}_p^{\gamma}(t^1\ -\ t^2; \theta^{\gamma,p})$$

$$cov\left(\kappa_p(h^1;\ t^1), \kappa_p(h^2;\ t^2)\right) = \ C_p^{\kappa}(h^1, h^2;\ t^1, t^2; \theta^{p})$$

$$= \ \tilde{C}_p^{\kappa,s}(d(h^1, h^2); \theta^{\kappa,s,p}) \cdot \tilde{C}_p^{k,t}(t^1\ -\ t^2; \theta^{\kappa,t,p})$$

In these covariance functions we have again assumed stationarity. We have developed a network structure customized distance metric for h in Equation (4.4), and we can still apply the same form of covariance function in Equation (4.6) for $cov\ (\beta_p(h^1), \beta_p(h^2))$. A number of temporal covariance functions has been developed in literature (see a summary in Cressie and Wikle, (2011)). For the spatio-temporal covariance function (Cressie and Wikle, 2011), we have assumed separability. Separability enables the spatio-temporal covariance matrix in the predictor to be computed as $\Sigma_p = \Sigma_p^{(s)} \otimes \Sigma_p^{(t)}$,and the inverse of the matrix as $\left(\Sigma_p\right)^{-1} = \left(\Sigma_p^{(s)}\right)^{-1} \otimes \left(\Sigma_p^{(t)}\right)^{-1}$, where $\otimes$ denotes the Kronecker product. In addition, to capture the temporal evolvement of the traffic flow and travel time, by employing this approach, we reduced the number of models to fit from $n \times |\mathcal{P}|$ to $|\mathcal{P}|$. Meanwhile, each of the model fitting process has increased complexity, induced by parameter estimation in additional covariance functions. Also, as we are considering the delay on each path $\tilde{\Psi}_p(\boldsymbol{h};\ t)$ instead of the delay on each path-time pair (p, i) as we did in the current approach, the number of training data (assumes that we are using the same number of runs of the DNL model as training data) will increase from K to K × n, where n is the time dimension (e.g., the number of time interval). When solving increasingly large-scale problem, the overall influence of model fitting complexity on the metamodeling computational efficiency is still to be seen. This problem may also alleviate by the dimension reduction approach introduced in the following chapter.

Chapter 6

Large-scale transportation networks

In this chapter we focus on large-scale transportation networks. Existing articles handle challenges brought by expanding scale of traffic network in different ways. Attacking a large-scale network using brutal force is usually unrealistic. For the same reason, aggregation, decomposition, other techniques are created to either reduce the network to a tractable scale, or simplify the modeling based on assumptions. In this chapter, we create a new version of network aggregation that solves large-scale network model.

Dimension reduction and improvement in model training speed

It is observed in the primary numerical examples that increase in size of network have a significant influence on the training time. Training time of the 501-paths Sioux Falls network is over a week. Although model training can be done offline, and is a reasonable investment on studying a large network considering the complexity of the delay operator, it is preferred in both analytical ease and practice to have better training efficiency and model simplicity when the network size grows very large. Possible approaches to reduce the model training time include employing better MLE algorithm and apply parallel computing, which requires access to great computational power.

In this section, we discuss another approach which tries to address the problem by reducing the dimension of the proposed Kriging metamodeling procedures. Currently the most time-consuming part of the model training is the computation of MLE (Equation (4.7)). An iterative pattern search algorithm is applied, which requires iteratively search and update each component of the parameter $\theta^{p,i} \in \mathbb{R}^{|\mathcal{P}|}$. The dimension of $\theta^{p,i}$, which equals the number of

paths in the network, significantly affect the speed of the algorithm and tractability of the model. To reduce the effect of network size $|\mathcal{P}|$ on the training speed and tractability, we proposed to redefine the distance metric of the Kriging model between two input departure rates h^1 and h^2. We start from partition the set of paths $\mathcal{P}$ into G groups, denoted by $1,\dots,G$,[15] based on similarities between paths. Paths that are more similar to each other will be assigned to the same group. We have

$$\mathcal{P} = \Gamma_1 \cup \Gamma_2 \cup \cdots \cup \Gamma_G$$

$$h_g^i = \left(w_i\left(h_{p_{j1}}^1 - h_{p_{j1}}^2\right)\right) \times \cdots \times \left(w_i\left(h_{p_{jl}}^1 - h_{p_{jl}}^2\right)\right) \quad \forall g = 1,\dots,G$$

where

$$\Gamma_g = \{p_{j1},\dots,p_{jl}\}$$

The new distance metric is then defined as follows

$$d_i(h^1,h^2) = \left(\left\|\left(h_g^{i,1} - h_g^{i,2}\right)\right\|_2^2 : g \in \mathcal{G}\right) \in \mathbb{R}^G \tag{6.1}$$

Comparing to the originally defined distance metric, this new definition reduces the dimension of d_i from number of paths $|\mathcal{P}|$ to number of path groups G. By reducing the number of components in the parameter, model flexibility is reduced. It is possible that the performance will suffer. However, we expect the scarifies will be low because as the number of components in the Kriging method grows to a very large number compared to the number of training data available, the MLE will be difficult to solve. This approach allows studying and flexibility in handling the tradeoff between training speed and model precision.

[15] Note that G: number of aggregated paths should not be confused with $\mathcal{G}$: set of all aggregated paths. The relationship between the two notion is $|\mathcal{G}| = G$

Clustering

Clustering is an important task in pattern recognition for data analysis. Various clustering techniques emerges in recent years in literature. The goal of clustering is to partition a set of examples into disjoint clusters based on some notion of (dis)similarity, such that related examples belong to the same cluster while unrelated examples belong to different clusters (Huang et al. 2014). Despite its apparent simplicity, it is not always clear how to select "related" examples because there are many possible ways of defining the similarity of examples for a given task, e.g., by using different similarity measures or distance metrics.

It is well known that using the Euclidean distance metric may not be a wise choice for a given task because it simply ignores the correlations between features, which usually contain useful discriminative information (Nguyen et al. 2016). In previous sections, we also discussed why Euclidean distance is replaced when we apply Kriging to the dynamic network loading (DNL) procedure. In principle, depending on the application domain, one wishes to learn a distance metric that satisfies some specific requirements. In our study, we have our defined dissimilarity on the domain of paths and extended distance metrics setup for the clustering tasks on domain of departure rates.

The following clustering methods are commonly seen in the literature:

- K-means
- Dbscan
- Support vector machine
- Random forest
- Other clustering methods

Two references to the clustering technique are the books by Alpaydin (2020) and Bishop (2006).

K-means clustering is one of the most popular and most efficient techniques for general clustering tasks. Typical applications include classification (Weinberger and Saul 2009; Faraki et al., 2018; Goldberger et al., 2005; Nguyen et al. 2018), regression(Nguyen et al., 2016), clustering(Jia et al. 2016; Wu et al. 2012; Bilenko et al. 2004; Yin et al. 2010), ranking(Lim and Lanckriet 2014), and kernel learning (Jain et al. 2012). In the next section, we will present how K-means can be used on extending the existing metamodeling approach to solve large-scale networks.

Framework of network aggregation

Path aggregation

Network aggregation can be seen in literature dating back to 1979 while early approach (Barton and Hearn, 1979) propose to aggregate links to form a reduce (sub)network in assisting network flow computation. Friesz (1985) provides a discussion on network aggregation emphasizing its application and opportunity found in the field of network equilibrium and network design.

To handle larger networks and dynamic versions, path clustering is a promising approach. Paths of similarities can be clustered based on features that induces sparsity in travel time and aggregated to form groups of paths that represents user choices.

In Chapter 5, we discuss several alternative choices of distance metrics in detail. They are recalled here to support our aggregation approach introduced in subsequent sections.

Creation of path groups

We define the similarity between path p and path q in this session as

$$\delta_{pq} = number\ of\ links\ shared\ by\ path\ p\ and\ q$$

Define the similarity of a group of paths G as

$$s_G = number\ of\ links\ shared\ by\ all\ paths\ in\ the\ group\ G$$

Aggregated path

To ease with conducting the learning method in a network with reduced dimension, we introduce a new type of path here.

Definition 6.1. (aggregated path) An aggregated path is defined based on a set of real paths in a network that shares a large number of links, or in other words, have high similarity. The (conceptual) aggregate path represents a group of paths to be considered as one choice for an individual traveler/user that has an expected travel time with some uncertainty brought by the differences in unshared links.

The uncertainty can be controlled and neglected when the paths are skillfully grouped, without harming the validity of the overall model.

We illustrate the idea of conceptual path using an example as follows:

Suppose we have a network of five paths: path #1, #2 and #3 go through a shared link a; paths #4 and #5 go through another shared link b. Then we have:

Path	Link	Shared link	User's perceived choice	Aggregated path
p_1	a, e	a	Choice 1	g_1
p_2	a, f			
p_3	a, g			
p_4	c, b	b	Choice 2	g_2
p_5	d, b			

Table **6-1** Path aggregation illustrative example: path information

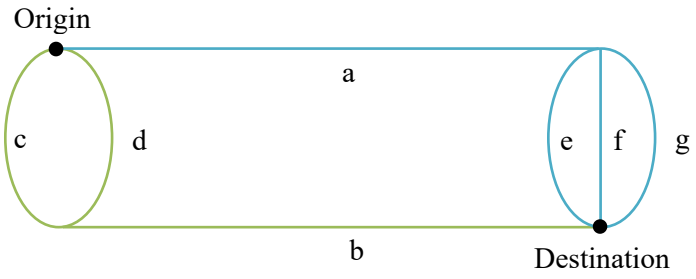

Figure **6-1** Path aggregation illustrative example: network

Explanation on intuition of the "aggregated path" is as follows: in real-world scenarios, the (conceptual) aggregated path usually complies with user's perceived choice: e.g. a choice between "going through highway" concept or "going through city-center" concept. Each choice not necessarily contains only one path, but more. Then during the decision-making process, each user does not necessarily choose among all possible paths, but among aggregated groups of paths that differs in important segment. The users could perceive shared links as crucial to determine the best choice. For example, a traveler that goes from west side of city Beijing to east side could face more than 20 possible distinct paths, but only two groups of choices: through highway, or through city center. This concept can be easily extended to other big cities around the world. In example illustrated in Table 6-1, aggregated path g_1 include 3 paths and g_2 include 2 paths. In real world large-scale network, it is reasonable to expect that through proper aggregation, each

aggregated path could include several to hundreds of real paths that distinguish themselves between each other only in segments that connects the shared links. This feature is the core to boost tractability.

Definition 6.2. (aggregated path flow) Denote g as an aggregate path on a set of paths $\{p_1, p_2, \dots p_m\}$ on a network. Then we have $h_g = \sum_{g \in \mathcal{G}} h_g$. In this equation, h_g is the aggregate path flow. Notation g represent both the "aggregated path flow" and the set of paths that is assigned to this aggregated flow.

The concept of path aggregation can be easily extended to "origin-destination" aggregation, where similar origins and destinations are aggregated to become aggregated origin and aggregated destination, respectively. In modeling based on the network, aggregated origins and destinations are considered instead. The idea is shown by an illustrative example network in the last section of this chapter.

Path aggregation would be relevant when working with large-scale network and networks with clear major-minor road structures. Computation on large-scale networks is difficult by traditional methods when original network is directly used. Dimension of path flow h will be huge, which directly challenge the computational limitations (software-specific limitation of variable dimension, computer memory capacities). Analytical methods for DTA also can scale badly and suffers from curse of dimensionality in scenarios of very large scale. By aggregating, the dimensionality issue can be reasonably overcome. For networks with a major-minor road structure, aggregation can provide a balancing between spatio (topological) simplicity and model accuracy. When properly aggregated, patterns in minor roads will be described in more simple way, and better granularity can be given to the major road that have significant influence on entire road network.

Framework and formulation of aggregation

We propose the following framework on aggregation path on the large-scale network

- Construction of the reduced network is to be noted in the following items:
    1. The aggregation paths will construct a “reduced network” from the original network.
    2. The aggregation paths will be the paths that the path based surrogate DNL model formulates on.
    3. Upper-level DUE can be either based on the reduced network or on the original network, with an accommodation of the path aggregation on delay operator. In reduced network, aggregated paths are treated as normal paths in formulation.
    4. Aggregated path departure rates and delays are generated (averaged) over represented paths on the network and the result is considered to be a reduced-scale counterpart of the original one.
    5. The aggregation paths need proper choice methodology: K-means clustering, ad hoc clustering, other aggregation methods.
    6. There needs to be a linked way between the path and aggregated paths that allocates DUE onto the large-scale network through aggregation. If stochastic Kriging is used, then we can take advantage of the stochasticity of surrogate model and define the output of original paths based on distribution/likelihood from an interval with mean and variance. If ordinary Kriging is used, then the reverse from reduced network to full network is based on the transition from the delays of the reduced network to its large-scale counterpart.
    7. Discussion on bound and error estimation (error rate).
- Delay operator on aggregated paths

Define Θ_g as delay on aggregated paths g, and Θ is the delay operator defined on the reduced network. $\hbar_g$ is the reduced-scale counterpart of h. Moreover, Θ is the reduced-scale counterpart of Ψ.

- Formulation of DUE on the reduced network based on aggregated path

Aggregation DUE of infinite dimension can be formulated by variational inequality (VI) as follows:

$$\left.\begin{array}{c} find\ \hbar^* \in \mathcal{L}\ such\ that \\ \sum_{g\in\mathcal{G}} \int_{t_0}^{t_f} \Theta_g(t,\hbar^*)\ \left(\hbar_g(t) - \hbar_g^*(t)\right) dt \geq 0 \\ \forall\ \hbar \in \mathcal{L} \end{array}\right\} VI\left(\Theta, \mathcal{L}, [t_0, t_f]\right)$$

where we have

$$\hbar = \left(\hbar_g : g \in \mathcal{G}\right)\ departure\ rates\ on\ aggragated\ paths\ g \in \mathcal{G}$$

$$\Theta = \left(\Theta_g : g \in \mathcal{G}\right) delay\ operator\ on\ aggregated\ path\ g \in \mathcal{G}$$

$\mathcal{L}$: set of all feasible aggregated departure rates

where $\mathcal{L}$ is the set of all feasible aggregated paths flows. The constraint follows aggregated flow conservation is given as:

Define $\boldsymbol{\hbar}$ as the finite dimensional departure rates on aggregated paths. Define $\mathcal{G}_{rs}$ as the set of aggregated paths between OD pair (r,s). Given the OD departure rate vector $R_{rs} = \left(R_{rs}^i : i = 1, \ldots\ldots, n\right)$for each OD pair $(r,s) \in \mathcal{W}$, we have the following constraints for finite dimensional aggregated path-specific departure rates $\boldsymbol{\hbar}_g$:

$$\Sigma_{g\in\mathcal{G}_{rs}}\ \boldsymbol{\hbar}_{g,i} = R_{rs}^i\ \forall\ i = 1, \ldots, n, \forall (r,s) \in \mathcal{W} \tag{6.2}$$

Same as the de-aggregated (original) network, we see that such constraints are decoupled for different time intervals *i* and different OD pairs (r,s). Therefore, the validity of the DUE analytical properties and solution procedures still hold.

Proof is trivial.

- Formulation of the DUE on original network based on path aggregation in delay operator

$$\left.\begin{array}{c} find\ h^* \in \Lambda\ such\ that \\ \sum_{p\in\mathcal{P}} \int_{t_0}^{t_f} \Psi_g(t,h^*) \sum_{p\in g} \left(h_p(t) - h_p^*(t)\right)\ dt \geq 0 \\ \forall\ h \in \Lambda \end{array}\right\} VI\left(\Psi, \Lambda, [t_0, t_f]\right)$$

where we have

$$h = (h_p : p \in \mathcal{P})$$

Algorithms of aggregation

This section introduces the algorithms for aggregating path on an arbitrary network. Number of aggregated paths (number of groups of paths) to be created can be predefined. This is useful in the cases that there is a target number of aggregated path (target size of the reduced network through aggregation). Alternatives are also given for cases such pre-definition is not needed.

Algorithm 1 A heuristic of path aggregation based on SLS

Suppose group $\mathcal{P}$ is the set of paths of an OD pair

1. Find matrix A of similarity
2. Starting from the smallest entries, which is number of links k any two paths shares, and group the paths that shares them into separate groups
3. Subtract k from all positive entries in matrix A
4. Go back to Step 2, until all entries in A are zero.

Algorithm 2 Grouping into predefined number of groups m

1. Find matrix A of similarity
2. Find the smallest m entries, and find the paths associates with the entries into m groups. Find the largest similarity k among the selected m entries.
3. Subtract k from all positive entries in matrix A
4. Go back to Step 2, until all entries in A are zero.

Algorithm based on K-means

K-means is a clustering method that can provide clusters based on similarities between items in a set. K-means algorithms interactively update the mean value and clustered items around the mean.

1.update mean value of path similarity

2.update clustered paths

3.check—go back to 1 if stopping criteria is not met

4. return clusters

K-means work with a "distance" between observations defined. In transportation network aggregation, observations are paths. The "distance" between any pair of paths p and q can be evaluated by the similarities defined in Chapter 5: SLS, DS, EDS (if applicable) and AS.

Comparison to support decision making between the three algorithms can be conduct on the following criteria: aggregation rate/efficiency and computational complexity.

Aggregation rate

For a large-scale network, define aggregation rate as the dissimilarity level between each two paths that are grouped into an aggregate path set.

Under the definition of a similarity matrix, the aggregation rate equals the smallest similarity between all paths in a path group.

The higher the aggregation rate is, the lower the dimension of the aggregated network is. We must make a tradeoff between dimension reduction and aggregation. Based on the definition of aggregation rate, we can control the degree of dimension reduction to best handle the large-scale network with least loss of accuracy during the reduction process.

Network aggregation by link

There are other types of aggregation schemes developed for handling large-scale networks discussed in the related literature. Barton and Hearn (1979) develops a decomposition method based on separable network structure and apply optimization problem on subnetwork and major network separately. The aggregation is conducted link-based and the reduced network is computed as a subproblem.

Numerical study framework

Picture of the network

The figure below shows a network that consist of a highway connecting two blocks of origins and destinations located on notes of lattice shaped road network. For ease of reference, we name this network Single Highway network.

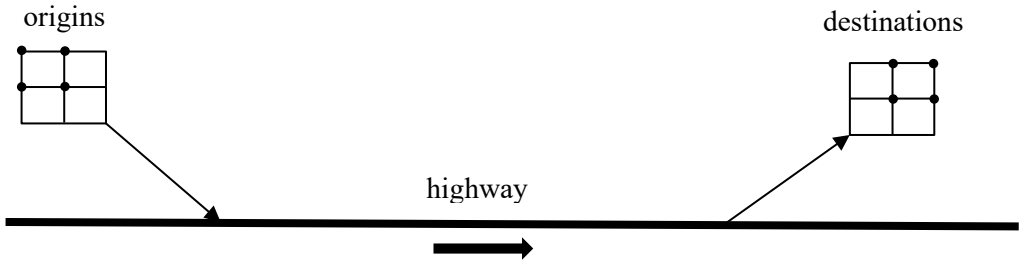

Figure **6-2** Single Highway network

Number of paths	196
Number of aggregated paths	16
Number of OD pairs	16

Table **6-2** Single Highway network data

The example is designed to show that path aggregation works on scenarios where the concept of some feasible path can be extracted by the crucial link it uses, in this example, the "highway" that connect all pairs of origins and destinations. All choices of paths only diverse from each other by choice of path within the lattice blocks from the origins to the highway entrance bridge, and from the highway exit bridge to the destinations. If the flows within the lattice structured subnetwork are not significantly changing, then these choices can actually be ignored in large-scale network calculation.

In this example, the rate of aggregation is

$$\frac{\text{Number of paths}}{\text{Number of aggregated paths}} = 12.25$$

Chicago sketch

This example is proposed to be revised for future research.

In this section, we test our proposed dimension reduction method in a real-world-scale Chicago Sketch testing network and apply the Kriging-based metamodeling method. The number of paths in the testing network is increased to 250,000. The result justifies the ability and potential of our proposed method in handling very large-scale problems efficiently.

We first employ the dimension reduction method discussed in previous Chapter 5 to reduce the dimension to a more tractable level.

In the Chicago Sketch network, we have a following data

Number of OD pairs	86,179
Number of paths	250,000
Average number of paths per OD pair	2.9

Table **6-3** Chicago Sketch network data

Chicago sketch has large number of OD pairs. In such cases, aggregating path will not be sufficient in reducing the network and improve tractability. To apply aggregation, the following decisions needs to be made to input into the aggregation algorithm:

- rate of path aggregation
- aggregation of OD pairs

Aggregation of OD pairs can be similarity deduced from the path aggregating algorithms. The detail is beyond the scope of this dissertation and is not discussed here in detail. In our design of numerical example, we include path aggregation to illustrate our proposed method.

We propose the following procedures to handle Chicago Sketch path aggregation for dynamic transportation models:

- finding distribution of path similarity
- decide the aggregation rate
- select aggregation algorithm
- conduct path aggregation
- transform the original problem (e.g., DUE, DNL) into its aggregated counterpart
- solve the aggregated problem
- interpret the result on either the aggregated or original problem via transformation

The numerical tests are beyond the scope of this dissertation and proposed to be done in future studies.

Chapter 7

Bi-level problems and MPEC

Introduction of bi-level problems

Bi-level models are an important technique of describing and analyzing dynamic transportation problems.

Categories of bi-level models:

- Network design problem
- Signal control
- Network capacity
- Tolling

Many bi-level problems are addressed in the form of a hierarchical optimization model or mathematical programming problem with equilibrium constraints (MPEC). Due to its multi-layer hierarchy, these problems are recognized as one of the most difficult and challenging problems in transportation systems management (Koh, 2007).

A hybrid model for dynamic user equilibrium (DUE) computation

In this section we propose a study on incorporating the dynamic network loading (DNL) metamodel to traditional DUE solution scheme to gain better computational efficiency with low loss of accuracy. Among different notions of DUE developed in the literature, we choose to start from the predictive continuous-time DUE, with SRDT choice (Friesz et al., 1993, 2011, 2013a). Because the concept of an embedded DNL procedure in all DUE problems is similar, the proposed framework can be extended to be applied to other types of DUE.

The formulation of DUE follows the one introduced in Chapter 2.

To facilitate further discussion on applying the DNL metamodel for SRDT DUE, we define the surrogate/approximate effective delay operator as follows:

$$\widetilde{\Psi}_p(t,h) \doteq \widetilde{D}_p(t,h) + \mathcal{F}(t + \widetilde{D}_p(t,h) - T_A)$$

where $\widetilde{D}_p(t,h)$ is the metamodel, or approximate delay operator, for the true/exact delay operator $D_p(t,h)$ learned by Kriging.

Fixed-point method with DNL metamodel

The fixed-point method introduced in Chapter 2 involves evaluation of the effective delay operator $\Psi_p(t,h^k)$ for a vector of path delay h^k in each iteration, which is one of the most time-consuming parts of the entire algorithm. A faster metamodel of the delay operator will significantly improve the computational efficiency of the problem. The computational improvement will be more significant considering that DUE is also widely used as a lower-level problem in many bi-level problems, where DUE solutions are usual computed multiple times to find an upper-level optimum.

The metamodeling presented in Chapter 4 is performed on finite dimensional counterpart of the infinite-dimensional delay operator. To apply our surrogate effective delay operator $\widetilde{\Psi}(h)$, we need to firstly define the approximate DUE with flexible temporal granularity (TG-DUE). We use the same notations as defined in Chapter 4.

Definition 7.1. (Dynamic user equilibrium with temporal granularity n (TG-DUE(n)) A vector of departure rates $h^* \in \Lambda$ is a dynamic user equilibrium with temporal granularity n if:

$\boldsymbol{h}^* \in \mathbb{R}^{n\times|\mathcal{P}|}$ is the finite n-dimensional counterpart of h^*

$\boldsymbol{\Psi}(\mathbf{h}^*)$ is the finite n-dimensional counterpart of $\Psi_p(t,h^*)$, and they satisfy

$$\boldsymbol{h}_{p,i}^{*} > 0, p \in \mathcal{P}_{rs} \quad \Rightarrow \quad \boldsymbol{\Psi}_{p,i}(\boldsymbol{h}^{*}) = v_{rs} \in \mathbb{R}_{++}$$

$$\forall i = 1, \dots, n \quad \forall (r,s) \in \mathcal{W}$$

We denote the dynamic user equilibrium defined this way by TG–DUE $(\Psi, \Lambda, [t_0, t_f], n)$.

For computation of the TG-DUE, we believe that a similar iterative solution method can be developed inspired by the iterative solution schemes for the original DUE problem, such as the fixed-point method. The solution method will still involve evaluation of the delay operator in each iteration. Assuming that we have successfully built the approximate surrogate $\tilde{\Psi}(h)$ from the metamodeling method in Chapter 2, there are several ways to employ the metamodel into the solution scheme of DUE: (1) to replace the effective delay operator by the surrogate model for all iterations in fixed-point method; and (2) to replace the effective delay operator by the surrogate model for some iterations, selected systematically. For the first look, approach (1) will gain the maximum computational saving for DUE computation. However, the error induced by using the surrogate model will possibly direct the solution trajectory away from the exact one, and results in more iterations than its original needed. Approach (2) tries to achieve best performance by keeping a balance between computational improvement and influence of approximation error to the trajectory, both brought by the surrogate model.

We note that similar approach as the second one can be extended to apply to any iterative solution schemes of DUE. For example, the proximal point method (PPM) developed by (Rockafellar, 1976) has also been applied to solving equilibrium problems with generalized monotonic operators. PPM evaluates the operator in each iteration. We propose to name any DUE computation method using both the exact delay operator/DNL procedures and the DNL metamodel a hybrid DUE solution scheme.

Future work in this approach includes (1) provide more rigorous definition and reformulation for the TG-DUE problem and study its properties; (2) develop one or more algorithms based on fixed-point or other existing DUE solution schemes; (3) develop a hybrid

DUE solution scheme based on the metamodel and items (1) & (2); (4) present numerical examples, compare DUE solution computed based on the surrogate delay operator with the exact DUE solution and discuss the findings.

Application of DNL metamodel on bi-level traffic models

Bi-level models are frequently discussed in dynamic traffic designs and managements. This class of problem assumes that travelers are responsive to the design or management of the network (e.g. tolls, signals, road conditions, speed limits) when each of them tries to minimize their perceived travel cost non-cooperatively. The notion of bi-level optimization models is closely related to Stackelberg games, where the central control authority of the upper level is the Stackelberg leader, and the travelers who respond to the leaders' decisions are Stackelberg followers.

These problems are typically modeled as mathematical program with equilibrium constraints (MPEC). The optimization framework has two levels: the upper-level optimizes system performance by searching for optimal traffic management actions, and the lower-level describes a traffic equilibrium model where travelers make adaptive travel choices as a result of the upper level decisions. Examples of bi-level optimization in transportation include studies on optimizing traffic signal control (Han et al., 2015d; Abdelfatah and Mahmassani, 1998; Ukkusuri et al., 2012; Chen and Ben-Akiva, 1998) and congestion pricing (Friesz et al., 2013b, 2007; Hearn and Yildirim, 2002) based on lower-level equilibrium traffic flow.

A common approach to compute an MPEC is to formulate the bi-level program into a mathematical program with complementarity constraints (MPCC), which reduces the problem into a single level. This approach relies on a complementarity system reformulation to the

variational inequality to represent the MPEC in a form of a "standard" nonlinear program. Limited by space, the formulation is not presented here.

In this section, we proposed to apply our DNL metamodel in MPEC solution scheme. We firstly introduce the formulation of MPEC using a bi-level signal control problem, following the assumptions and settings in (Han et al., 2015d), and then provide description and discussion on our proposed ideas.

A bi-level signal control problem

Following Han et al., (2015d), we consider an upper-level problem with decision variables as signal green splits, and lower-level problem as a DUE. We denote by the vector η(t) the possibly time-varying signal green splits associated with the entire network. We note that the signal decisions $\eta(t)$ are embedded within the DNL subproblem of the lower-level DUE model. The dependence of the DNL procedure on these signal splits is indicated in the following notation:

$$\Psi(t, h; \eta) = \left(\Psi_p(t, h; \eta), p \in \mathcal{P}\right) \qquad \eta(t) \in Y$$

where Y denotes the set of feasible signal splits.

Using variational inequality formulation for the DUE presented above, we can formulate the lower-level DUE problem with a given signal timing $\eta(t)$ as follows:

$$\left.\begin{array}{c} find\ h^* \in \Lambda\ such\ that \\ \sum_{p\in\mathcal{P}} \int_{t_0}^{t_f} \Psi_p(t, h^*; \eta)\ \left(h_p(t) - h_p^*(t)\right) dt \geq 0 \\ \forall\ h \in \Lambda \end{array}\right\} VI\left(\Psi, \Lambda, \eta, \left[t_0, t_f\right]\right) \tag{7.1}$$

where Λ is the feasible region for h, defined as

$$\Lambda = \left\{ h(\cdot):\ h_p(t) \geq 0,\ \sum_{p\in\mathcal{P}_{rs}} h_p(t) = R_{rs}(t) \qquad \forall t \in \left[t_0, t_f\right],\ \forall (r,s) \in \mathcal{W} \right\} \tag{7.2}$$

The upper-level objective is to minimize the total travel costs on the entire network, by finding a set of optimal signal controls $\eta(t)$. The upper-level's objective function is formulated as follows:

$$\min_{\eta(t)\in Y} \sum_{p\in\mathcal{P}} \int_{t_0}^{t_f} \Psi_{\mathrm{p}}\,(t, h^*; \eta) h_p^*(t) dt$$

where h^* is the solution to the lower-level DUE problem, formulated as a VI, in (7.1). Therefore, we have the following MPEC formulation for the entire bi-level dynamic signal control problem:

$$\min_{\eta(t)} \sum_{p\in\mathcal{P}} \int_{t_0}^{t_f} \Psi_{\mathrm{p}}\,(t, h^*; \eta) h_p^*(t) dt$$

$$subject\ to \begin{cases} \sum_{p\in\mathcal{P}} \int_{t_0}^{t_f} \Psi_{\mathrm{p}}\,(t, h^*; \eta)(h_p(t) - h_p^*(t))\, dt \geq 0 \quad \forall h \in \Lambda \\ h^* \in \Lambda, \qquad \eta(t) \in Y \end{cases}$$

The MPEC problem is a very difficult problem to solve because it is non-convex, has a semi-infinite constraint (the VI), and there is very little regularity for the delay operator $\Psi_{\mathrm{p}}(t, h^*; \eta)$ regarding its arguments and parameters. Having the delay operator metamodeling framework, we proposed the following approaches:

1. Develop DNL metamodel for the delay operator with dependency on signal control parameters and obtain the closed-form approximate delay operator $\widetilde{\Psi}_p\,(t, h^*; \eta)$
2. Define and reformulate the MPEC problem with TG-DUE as the lower-level problem
3. Apply the closed-form $\widetilde{\Psi}_p\,(t, h^*; \eta)$ into analytical investigation and efficient calculation of MPEC problems. This includes two possible approaches:

   (a) Replace the original non-closed form delay operator with closed form $\widetilde{\Psi}_p\,(t, h^*; \eta)$ in all mathematical formulations. Because the $\widetilde{\Psi}_p\,(t, h^*; \eta)$ enjoys many analytical properties, such as continuity and differentiability that the original delay

operator does not, we can (1) perform analytical investigations that would not be possible before, and (2) reformulate the MPEC into MPCC and solve the single-level problem by existing algorithms/commercial software.

(b) Apply heuristic methods to solve the formulated MPEC and use the hybrid DUE solution scheme discussed in the section about the fixed-point method with DNL metamodel to improve the solution efficiency of the entire problem. Because all heuristic methods that tries to optimize the upper-level variables will need to repeatedly compute lower-level DUE solutions, the computational saving of using the metamodel will be even more significant.

Chapter 8

Discussion

In this dissertation we discuss an interdisciplinary research that studies statistical learning on dynamic transportation models. In the previous chapters, we introduced models based on Kriging, NNs, Neural, and clustering on single-level and bi-level DTAs. In this chapter we will discuss work that has been presented, application of the developed method, significance, limitations, and directions of future work.

This dissertation contains three main parts. We started from the most general categories of dynamic transportation models and discuss the dynamic structure and modeling approaches developed in the literature for the DTA branch in depth. Among all dynamic models the author of this dissertation reviewed in detail the DTA in particular. This review can be found in Chapters 1 and 2. Through this branch of research, we identified the dynamic network loading (DNL) submodel as a breakthrough point of applying statistical learning to dynamic transportation models. The second and third part of the dissertation discussed this novel scheme of metamodeling applied on the DNL submodel and extensions, including extending the distance metric and extending the scale onto large-scale (real-world size) networks. Why we choosing Kriging and how we applied it are discussed in depth in Chapter 3 and Chapter 4.

In Chapter 4, this dissertation proposed a novel metamodeling approach based on statistical learning for a class of DNL models. The goal was to provide a class of surrogate DNL models that approximated the exact ones, with considerable benefits including closed-form representation, improved regularity, and superior computational efficiency, at the expense of minor, yet controllable approximation errors. To achieve this, we employed the Kriging method to interpret and approximate the inherent input-output mechanism of the delay operator. This

method considers the observed functional relationship as a realization of a GRF, and the Kriging estimate corresponds to the posterior predictive density of the function approximation.

Within the framework of metamodeling, we first articulated the precise definition of delay operators in continuous time and then defined its discrete counterpart in finite dimensional spaces. Then we implemented statistical learning methodology by utilizing the network structure and path information to perform Kriging on a non-conventional space with network- specific distance metric. Training of the model parameters was formulated as an MLE based on the Gaussian process, and it yielded a closed-form predictor. Following this, a way to systematically generate training data based on LHD was proposed and complemented by a second one based on a heuristic and adaptive approach. This metamodeling approach produced, as its output, a family of surrogate DNL models that approximated the exact ones. Compared to the delay operators obtained from conventional DNL procedures, the proposed ones enjoy much-improved analytical properties that benefited DTA analysis and computation significantly; these included closed-form expression, (Lipschitz) continuity, differentiability, and closed-form expression for the Jacobians. These advantages are partially discussed in this chapter, and further applications are proposed in future studies. On the computational side, it was shown in our numerical study that the surrogate DNL models have far-superior computational efficiencies and were 9 to 455 times faster than conventional ones. Moreover, the approximation errors remained low (less than 8%) throughout our numerical experiment.

The occurrence of vehicle spillback may cause the delay operator to be discontinuous (Song et al., 2018; Han et al., 2016a; Szeto, 2003). Such a fact did not render our metamodel ineffective, because Kriging is a highly flexible metamodeling method, in that it can fit functions/mappings with different degrees of smoothness including discontinuities. However, the discontinuity in the delay operator could affect the error estimate of the Kriging method. Further study is needed to distinguish the performance of Kriging by different regularities of the delay

operator (i.e., the spillback versus non-spillback cases). The proposed statistical metamodeling approach opened the pathway to a family of new network performance models with tractability barely seen in existing ones.

We noted that there are many other advanced experimental design methods that could potentially further improve the performance of our metamodels. These include the penalty method for solving the likelihood maximization problem (Li and Sudjianto, 2005) and dimension reduction techniques based on the clustering of the paths.

Chapter 5 develops dimension reduction techniques and extends the distance metric, which was crucial in determining the proper function of the Kriging metamodel into multiple categories and forms to enable an increased range of problems to be eligible for use. Moreover, the large-scale methods proposed and articulated were brand new in the area of statistical large-scale modeling of DUE. DUE models, as widely found, are not tractable because the problem grows with network size. The explosion of number of paths and OD pairs and the correlation made the problem non-trivial. These problems were from real-world size networks and closely linked to practical problem solving. They were also difficult to describe using linear and tractable methods, losing details if they are reduced too much and over-fitting if they are approximated using inappropriately-simple models. In Chapter 6, we presented in particular the concept of dimension reduction rate to devise the proper choice of reducing the level from the original network to the one that is being modeled. The process held the model in an appropriate level of details while it sufficiently reduced the computational burden to a tractable degree.

In Chapter 6 and 7 the author explores the impact of our proposed study on DTA by

- extending the method to large-scale networks through network aggregation
- incorporating the surrogate DNL models into the computation of DUE and DSO problems; and

- devising gradient-based methods for solving bi-level problems on large-scale networks

With the presented framework, proposed future research includes testing the possibility of incorporating real-world measurements into the delay operator through the opened path that links statistical learning with dynamic transportation models.

Significance

Metamodeling provided a new perspective in how transportation problems can benefit from the latest developments of statistical learning. Statistical learning, which has widely-recognized advantages, can help the transportation model in overcoming the curse of dimensionality and reduce the obstacles obstacle large network analysis imposed by computational cost. The proposed method smoothed the mathematical formulation of traditionally non-smooth and non-differentiable operator.

Large amount of literature showed transportation problems can be solved mathematically when properly reformulated and expressed. The real-world transportation systems continue to evolve and attract scientists' interest with its dynamic nature and increasing complexity, especially in economically fast-developing regions. Layers of socio-economic meanings, factors of environmental influences, and big data supported technology are emerging in modern transportation systems. The design and management of modern networks is becoming increasingly advanced and data supported. With the thriving of smart phone applications and big data behind the flow of users on any road, we are better positioned than any time in the history to study the mystery of modern networks. Meanwhile, traditional issues, such as traffic jams and bottlenecks, are never completely eliminated in most of the modern networks and are expanding

beyond the traditional scope, now to be seen in both large cities and small towns increasingly. This phenomenon poses a great challenge to the experts who wish to improve it.

In this dissertation, we built a new pathway from the transportation networks to statistical learning. The challenging network research in shown to benefit highly from the new models developed from powerful statistical learning tools. Without the condition of degree of differentiability and analytical properties, multiple types of DNL submodel of multiple kinds and a wide range of scales, either existing or forthcoming, can be freely modeled by the developed Kriging model with reasonable error and high tractability.

In this dissertation, we propose a new method for tackling this important and difficult issue by providing a class of surrogate DNL models based on a statistical learning method known as Kriging. This work forms concrete foundation of our proposed family of modeling techniques. We present a metamodeling framework that systematically approximates DNL models and is flexible in the sense that it enables the modeler to make trade-offs between model granularity, complexity, and accuracy. It is shown that such surrogate DNL models yield highly accurate approximations (with error rates below 8%) and superior computational efficiency (9 to 455 times faster than conventional DNL procedures, such as those based on the link transmission model). Furtherly, we presented that the developed DNL surrogate model can smoothly replace traditional ones in a typical DUE formulation and enable the formulation to enjoy closed-form analytical delay operators and much better analytical properties: Lipschitz continuity, infinitely differentiability and closed-form Jacobians.

Summary

This dissertation consisted of eight chapters, discussing interdisciplinary research between two main fields: dynamic transportation models and statistical learning. The dissertation described and discussed comprehensively projects of this doctoral study.

We started with a comprehensive introduction to the two main areas of interest in this dissertation: dynamic transportation models, and statistical learning. Building linkage between the two fields yielded our new interdisciplinary study that absorbed the advantages and strengths of both fields. The mathematical formulation and properties of DTA models were presented in detail and discussed, and the platform of metamodeling that facilitates a new way of modeling the problem was built systematically. The literature review on dynamic transportation models, elaborated on DTA specifically. Following that, we introduced statistical learning and metamodeling concepts. The connection between the two fields was built on the project of applying Kriging onto DNL metamodeling in Chapter 3. One sentence here to explain the connection.

- *Statistical metamodeling of DNL*

Statistical metamodeling of the DNL connect statistical learning to dynamic transportation models. We use Kriging to provide surrogate DNL model that works for analytical DUE and bi-level DTA. We have utilized advanced experimental design for training and testing process. Result shows significantly improve in computational efficiency and model tractability. Kriging is an exact interpolation method and flexible statistical learning tool in practice. These features are clearly reflected in the positive result in numerical examples, promising its value in furtherly pushing boundaries of DTA modeling and computations. Through our work, available classes of DNL models are extended from non-close form to close form ones. Future directions

include testing and investigating the analytical advances it brings to DUE computation and DTA studies.

- *Distance metrics*

Distance metrics is of great importance in determining how the Kriging model will perform. In this dissertation, we extended the distance metric from Euclidean distance in Ordinary Kriging to non-Euclidean metrics to account for the network structure imposed cross-paths effects and interactions. The introduction of non-Euclidean metrics into the metamodeling framework facilitated the use of Kriging and all distance based statistical learning methods (including some popular clustering algorithms) on problems and cases where the input is correlated by non-Euclidean (dis)similarities. The Kriging method was proven effective by numerical experiments in replacing the traditional DNL models for improved computational efficiency with a very small error rate. A direction worth investigating is whether the defined distance in our model is a valid metric. We have provided proof and discussion on the validity of the distance we developed.

- *Large-scale network*

To deal with the challenges brought by the high dimensionality, we developed a dimension reduction scheme to reduce computational burdens that analyzing large-scale networks brought. Path aggregation, reduced network, and a new framework for DUE on a large-scale network with aggregated DNL was presented and discussed. The framework has inherited the analytical form and properties of the standard VI formulation of a continuous-time DUE problem, enabling the application of algorithms with a differential delay operation on large-scale networks. This approach expanded the scope of use of the statistical learning-based approach. New testing networks of more than 250,000 paths are proposed/forthcoming in future research.

- *Applications*

We discuss two categories of applications

1. application of DNL metamodel/surrogate model to DUE problem;

2. application of DNL metamodel/surrogate model to bi-level dynamic transportation problems.

In conclusion, our study contributes to the DNL and DTA research both analytically and computationally. The proposed and studied scheme forms a brand-new approach of working with the challenges dynamic transportation models bring. Numerical studies show promising computational advances compared to the traditional procedures, and discussion on analytical features of the method that we propose is promising. Our approaches involved concepts and techniques in multiple areas of scientific research, including mathematical programming, statistical learning, and metamodeling, DTA, and differential games.

Future directions of this research would first include research on in-depth interpretation on traffic pattern identification Kriging metamodel enables. On the traffic management side, quantitative insights can be draw from the learned surrogate model and evaluation of a traffic management scheme based on surrogate model are promising directions.

Appendix A

Mathematical proof on validity of distance metric

A distance metric $d(x_1, x_2)$ is a valid metric measure if it satisfies

1. Symmetry: $d(x_1, x_2) = d(x_2, x_1)$
2. Positivity: $d(x_1, x_2) \geq 0$
3. Non-degeneracy: $d(x_1, x_2) = 0 \; if \; and \; only \; if \; x1 = x2$
4. Triangle inequality: $d(x_1, x_2) \leq d(x_1, x_3) + d(x_3, x_2)$

Additionally, the distance metric $d(x_1, x_2)$ is translation invariant if it satisfies $d(x_1 + \alpha, x_2 + \alpha) = d(x_1, x_2)$. A distance metric $d(x_1, x_2)$ is a pseudo-metric if requirements 1 and 2 are satisfied and requirement 3 is partially satisfied (the "if" is satisfied)

The proof validity of our proposed distance metric based on proposed similarity between paths is given here.

Propose distance metric for all $p \in \mathcal{P}, 1 \leq i \leq n$ as follows

$$\boldsymbol{d}_{p,i} = \left(\delta_{pq} \left\| w^i \circ (\boldsymbol{h}_q^1 - \boldsymbol{h}_q^2) \right\|_2^2 : q \in \mathcal{P}\right) \in \mathbb{R}^{|\mathcal{P}|}$$

In the following proof, we neglect all subscript (p, i) of d for notation simplicity. This does not influence the mathematical validity of the proof.

$$d(h^1, h^2) = \left(\delta_{pq} \left\| w^i \circ (\boldsymbol{h}_q^1 - \boldsymbol{h}_q^2) \right\|_2^2 : q \in \mathcal{P}\right)$$

$$d(h^1, h^2) \in \mathbb{R}^{|\mathcal{P}|}$$

1. Symmetry

$$d(h^1, h^2) = \left(\delta_{pq} \left\| h_q^1 - h_q^2 \right\|_2^2 : q \in \mathcal{P}\right)$$

$$d(h^2, h^1) = \left(\delta_{pq} \left\| h_q^2 - h_q^1 \right\|_2^2 : q \in \mathcal{P}\right)$$

Because

$$\| h_q^1 - h_q^2 \|_2^2 = \| h_q^2 - h_q^1 \|_2^2 \ \forall\, q \in \mathcal{P}$$

We have

$$d(h^1, h^2) = d(h^2, h^1)$$

2. Positivity

$$d(h^1, h^2) = \left(\delta_{pq} \| h_q^1 - h_q^2 \|_2^2 : q \in \mathcal{P}\right)$$

We have

$$\delta_{pq} \geq 0 \ based\ on\ definition$$

$$\|h_q^1 - h_q^2\|_2^2 \geq 0 \ based\ on\ property\ of\ norm$$

Therefore

$$d(h^1, h^2) \geq 0$$

3. Positive definiteness

The positive definiteness depends on δ_{pq}. In this proof, we assume that the set of paths on the network fulfills $\delta_{pq} \neq 0$:

Proof of "if"

Suppose $h^1 = h^2\ then\ we\ have\ h_q^1 = h_q^2\ \forall\, q \in P$

Therefore

$$||h_q^1 - h_q^2||_2^2 = 0\ \forall q \in P$$

Then we have

$$d(h^1, h^2) = \mathbf{0}$$

Proof of "only if"

Suppose

$$d(h^1, h^2) = \mathbf{0}$$

Then we have

$$\delta_{pq} \left\| h_q^1 - h_q^2 \right\|_2^2 = 0 \quad \forall\, q \in P$$

$$\delta_{pq} \perp \left\| h_q^1 - h_q^2 \right\|_2^2 \quad \forall q \in \mathcal{P}$$

If δ_{pq} is non-zero in all entries, we will have $\left\| h_q^1 - h_q^2 \right\|_2^2 = 0 \quad \forall\, q \in P$

Therefore, based on the property of norm, we have

$$h_q^1 = h_q^2 \quad \forall\, q \in P$$

We have

$$h^1 = h^2$$

Under the condition: $\delta_{pq} \neq 0\ \forall p, q \in \mathcal{P}$, the distance metric is valid.

We note that $\delta_{pq} \neq 0$ may not hold for all networks. In those cases, the validity can be ensured by simply introducing a very small positive parameter ϵ added to the original δ_{pq} definition. For example, the definition of δ_{pq} can be revised as

$$\delta_{pq} = \begin{cases} \Sigma ca, & if\ ca \neq 0\ for\ p\ and\ q \\ \epsilon & otherwise \end{cases}$$

here ϵ is a very small positive number.

In the case that δ_{pq} is given as its original form, which is nonnegative and not adjusted with the above method, the distance metric we propose is a valid pseudo-metric. This result holds for any network and all δ_{pq}. In some networks cases that the network structure satisfies $\delta_{pq} > 0$ requirements, this distance metric is also a valid metric measure.

4. Triangle inequality

$$d(h^1, h^2) = \left(\delta_{pq} \left\| h_q^1 - h_q^2 \right\|_2^2 :\ q \in \mathcal{P}\right)$$

$$d(h^1, h^3) = \left(\delta_{pq} \left\| h_q^1 - h_q^3 \right\|_2^2 :\ q \in \mathcal{P}\right)$$

$$d(h^3, h^2) = \left(\delta_{pq} \left\| h_q^3 - h_q^2 \right\|_2^2 :\ q \in \mathcal{P}\right)$$

$$d(h^1, h^3) + d(h^3, h^2) = \left(\delta_{pq} \left\| h_q^1 - h_q^3 \right\|_2^2 :\ q \in \mathcal{P}\right) + \left(\delta_{pq} \left\| h_q^3 - h_q^2 \right\|_2^2 :\ q \in \mathcal{P}\right)$$

$$= \left(\delta_{pq}(\| h_q^1 \ - h_q^3 \|_2^2 + \| h_q^3 \ - h_q^2 \|_2^2) : \ q \ \in \ \mathcal{P}\right)$$

Because we have the following:

$$\| h_q^1 - h_q^2 \|_2^2 = \| h_q^1 - h_q^3 + h_q^3 - h_q^2 \|_2^2 \leq \| h_q^1 - h_q^3 \|_2^2 + \| h_q^3 - h_q^2 \|_2^2$$

5. Translation invariance

We have

$$d(h^1 + \alpha, h^2 + \alpha) = \left(\delta_{pq} \| h_q^1 + \alpha \ - h_q^2 + \alpha \|_2^2 : \ q \ \in \ \mathcal{P}\right)$$

Because

$$\| h_q^1 + \alpha \ - h_q^2 + \alpha \|_2^2 = \| h_q^1 \ - h_q^2 \|_2^2$$

We have

$$d(h^1 + \alpha, h^2 + \alpha) = \left(\delta_{pq} \| h_q^1 - h_q^2 \|_2^2 : \ q \ \in \mathcal{P}\right) = d(h^1, h^2)$$

With the complete proof containing items 1 to 5, we verify that the proposed metric is valid under the requirements over all paths and time in all OD pairs.

Appendix B

Mathematical proof on validity of covariance function

A valid covariance function is a positive definite function, and vice versa. In this appendix we give a brief proof that our (newly) customized distance metric does not change the validity (the positive definiteness) of the covariance function. For each pair of (p, i), the distance metric is

$$\boldsymbol{d}_{p,i} = \left(\delta_{pq} \left\| w^i \circ (\boldsymbol{h}_q^1 \quad - \boldsymbol{h}_q^2) \right\|_2^2 : q \in \mathcal{P}\right) \in \mathbb{R}^{|\mathcal{P}|}. \tag{A.1}$$

and the covariance function is

$$\begin{aligned} \mathcal{C}(\boldsymbol{d}(\boldsymbol{h}^1, \boldsymbol{h}^2), \theta) &= \sigma^2 \exp\left(-\boldsymbol{d}_{p,i}(\boldsymbol{h}^1, \boldsymbol{h}^2)^T \cdot \theta\right) \\ &= \sigma^2 \exp\left(- \left(\delta_{pq} \left\| w^i \circ (\boldsymbol{h}_q^1 \quad - \boldsymbol{h}_q^2) \right\|_2^2 : q \in \mathcal{P}\right)^T \cdot \theta\right). \end{aligned}$$

Our metamodeling direction is fundamentally based on the newly developed distance metric (A.1). We note that change in the distance metric will change the underlying space the covariance function is defined on. It is the point why we are having this proof, because changing in the underlying space will possibly affect the property of the covariance function. In this proof, we are going to show that the exponential form of covariance remains to be valid, when distance is changed to (A.1).

It is to be noted that the model is extended in Chapter 5. In this proof we take all the proposed extensions on distance metric into consideration. We recall that δ_{pq} is a scalar that indicates the similarity between two paths p and q. The following proof only uses the property that δ_{pq} is a scalar, and therefore it is applicable for any generalized form of path similarity.

Denote $\boldsymbol{d}^E$ as the Euclidean distance. We have

$$\boldsymbol{d}^E(\boldsymbol{h}^1, \boldsymbol{h}^2) = \left(\left\| (\boldsymbol{h}_q^1 \quad - \boldsymbol{h}_q^2) \right\|_2^2 : q \in \mathcal{P}\right)$$

Consider the (standard) exponential covariance function with Euclidean distance

$$\mathcal{C}^E(\boldsymbol{d}^{\boldsymbol{E}}(\boldsymbol{h}^1,\boldsymbol{h}^2),\theta)=\sigma^2\exp(-\boldsymbol{d}^E\cdot\theta)=\sigma^2\exp\left(-\left(\left\|\left(\boldsymbol{h}_q^1\ \ -\boldsymbol{h}_q^2\right)\right\|_2^2 : q\ \in\mathcal{P}\right)^T\cdot\theta\right)$$

The exponential covariance function is among the most common form of covariance function used in Kriging, and it is verified to be positive definite when Euclidean distance is used. This means on the entire vector space of the input departure rates, $\mathcal{C}^E(\boldsymbol{d}^{\boldsymbol{E}}(\boldsymbol{h}^1,\boldsymbol{h}^2),\theta)$ is a valid covariance function, i.e.

$$\sum_{j,k=1}^{n} a_j a_k \sigma^2 \exp\left(-\left(\left\|\left(\boldsymbol{h}_q^j\ \ -\boldsymbol{h}_q^k\right)\right\|_2^2 : q\ \in\mathcal{P}\right)^T\cdot\theta\right)>0\ \ \forall a_j,a_k\in\mathbb{R}\ \ \forall \boldsymbol{h}^j,\boldsymbol{h}^k\in\mathbb{R}_+^{n\times|P|} \tag{A.2}$$

This is the positive definiteness of a valid covariance function. Recall that the vector of departure rates is defined as

$$\boldsymbol{h}=(\boldsymbol{h}_p : p\in\mathcal{P})\in\mathbb{R}_+^{n\times|P|}$$

For each pair of (p,i), our proposed covariance function for the metamodel is

$$\mathcal{C}(\boldsymbol{d}(\boldsymbol{h}^1,\boldsymbol{h}^2),\theta)=\sigma^2\exp\left(-\boldsymbol{d}_{p,i}(\boldsymbol{h}^1,\boldsymbol{h}^2)^T\cdot\theta\right)$$

$$=\sigma^2\exp\left(-\left(\delta_{pq}\left\|\,w^i\circ\left(\boldsymbol{h}_q^1\ \ -\boldsymbol{h}_q^2\right)\right\|_2^2 : q\ \in\mathcal{P}\right)^T\cdot\theta\right)$$

This covariance function uses our newly developed distance function. We then observe that the change in distance function can be equivalently represented by a modification on the inputs. Let

$$\overline{\boldsymbol{h}_p}=\sqrt{\delta_{pq}}\,w^i\circ\boldsymbol{h}_p \tag{A.3}$$

and

$$\overline{\boldsymbol{h}}=\left(\overline{\boldsymbol{h}_p} : p\in\mathcal{P}\right).$$

Using $\overline{\boldsymbol{h}}$**,** our proposed covariance function can be rewritten. We have

$$\mathcal{C}(\boldsymbol{d}(\boldsymbol{h}^1,\boldsymbol{h}^2),\theta)=\sigma^2\exp\left(-\left(\delta_{pq}\left\|\,w^i\circ\left(\boldsymbol{h}_q^1\ \ -\boldsymbol{h}_q^2\right)\right\|_2^2 : q\ \in\mathcal{P}\right)^T\cdot\theta\right)$$

$$\sigma^2 \exp\left(-\left(\left\|\sqrt{\delta_{pq}}\ w^i \circ \boldsymbol{h}_q^1 - \sqrt{\delta_{pq}}\ w^i \circ \boldsymbol{h}_q^2\right)\right\|_2^2 : q \in \mathcal{P}\right)^T \cdot \theta\Big)$$

$$= \sigma^2 \exp\left(-\left(\left\|\left(\overline{\boldsymbol{h}_q^1} - \overline{\boldsymbol{h}_q^2}\right)\right\|_2^2 : q \in \mathcal{P}\right)^T \cdot \theta\right)$$

By rewriting the covariance, we just need to prove that

$$\sum_{j,k=1}^{n} a_j a_k \sigma^2 \exp\left(-\left(\left\|\left(\overline{\boldsymbol{h}_q^j} - \overline{\boldsymbol{h}_q^k}\right)\right\|_2^2 : q \in \mathcal{P}\right)^T \cdot \theta\right) > 0 \ \forall a_j, a_k \in \mathbb{R} \ \forall \ \overline{\boldsymbol{h}^j}, \overline{\boldsymbol{h}^k} \in \mathbb{R}_+^{n\times|P|} \tag{A.4}$$

to prove that the proposed covariance function is valid. The remaining of the appendix proves the property shown in (A.4). We start from the relationship between $\overline{\boldsymbol{h}_p}$ and $\boldsymbol{h}_p$. Because (A.3) only involves elementary multiplications on vectors and scalars, we have

$$\overline{\boldsymbol{h}^j}, \overline{\boldsymbol{h}^k} \in \mathbb{R}_+^{n\times|P|} \quad \forall \boldsymbol{h}^j, \boldsymbol{h}^k \in \mathbb{R}_+^{n\times|P|}$$

We note that the property in (A.1) is satisfied for any choice of vector pair that is in the vector space $\mathbb{R}_+^{n\times|P|}$. Since $\overline{\boldsymbol{h}^j}, \overline{\boldsymbol{h}^k} \in \mathbb{R}_+^{n\times|P|}$, we can derive that (A.2) is satisfied for all or all $\overline{\boldsymbol{h}^j}, \overline{\boldsymbol{h}^k}$. Plugging in any $\overline{\boldsymbol{h}^j}, \overline{\boldsymbol{h}^k} \in \mathbb{R}_+^{n\times|P|}$ into (A.2), we will have

$$\sum_{j,k=1}^{n} a_j a_k \sigma^2 \exp\left(-\left(\left\|\left(\overline{\boldsymbol{h}_q^j} - \overline{\boldsymbol{h}_q^k}\right)\right\|_2^2 : q \in \mathcal{P}\right)^T \cdot \theta\right) > 0 \ \forall a_j, a_k \in \mathbb{R}$$

This can be directly rewritten as

$$\sum_{j,k=1}^{n} a_j a_k \sigma^2 \exp\left(-\left(\left\|\left(\overline{\boldsymbol{h}_q^j} - \overline{\boldsymbol{h}_q^k}\right)\right\|_2^2 : q \in \mathcal{P}\right)^T \cdot \theta\right) > 0 \ \forall a_j, a_k \in \mathbb{R} \ \forall \overline{\boldsymbol{h}^j}, \overline{\boldsymbol{h}^k} \in \mathbb{R}_+^{n\times|P|}$$

This is the positive definiteness property, and it shows that the covariance function we use is valid. With the proof, we verify that the covariance function used is valid, in particular when the customized distance metric (A.1) is used. The result hold for both standard and extended model over all paths and time in all OD pairs.

Reference

Amin, S. M., Rodin, E., Liu, A.-P., Rink, K., and García-Ortiz, A., 1998. Traffic prediction and management via rbf neural nets and semantic control. Computer-Aided Civil and Infrastructure Engineering, 13(5): 315–327.

Abdelfatah, A., and Mahmassani, H., 1998. System optimal time-dependent path assignment and signal timing in traffic network. Transportation Research Record: Journal of the Transportation Research Board, (1645), 185–193.

Alpaydin, E., 2020. Introduction to machine learning. MIT press.

Axhausen, K.W., Horni, A. and Nagel, K., 2016. The multi-agent transport simulation MATSim (p. 618). Ubiquity Press.

Barton, R. R., and Hearn, D. W., 1979. Network aggregation in transportation planning models (No. DOT-TSC-RSPA-79-18). United States. Dept. of Transportation. Research and Special Programs Administration.

Bliemer, M., 2007. Dynamic queuing and spillback in analytical multiclass dynamic network loading model. Transportation Research Record: Journal of the Transportation Research Board 2029, 14–21.

Raadsen, M. P., Bliemer, M. C., and Bell, M. G., 2016. An efficient and exact event-based algorithm for solving simplified first order dynamic network loading problems in continuous time. Transportation Research Part B: Methodological, 92, 191-210.

Bilenko, M., Basu, S., and Mooney, R. J., 2004. Integrating constraints and metric learning in semi-supervised clustering. Proc. 21st Int. Conf. Mach. Learn., pp. 81–88.

Bishop, C.M., 2006. Pattern recognition and machine learning. Springer.

Boyce, D., Lee, D.H., Ran, B., 2001. Analytical models of the dynamic traffic assignment problem. Networks and Spatial Economics 1:377–390

Cascetta, E. and Cantarella, G.E., 1993. Modelling dynamics in transportation networks: State of the art and future developments. Simulation practice and theory, 1(2), pp. 65–91.

Celikoglu, H. B., 2006. Application of radial basis function and generalized regression neural networks in non-linear utility function specification for travel mode choice modelling. Mathematical and Computer Modelling, 44(7), 640–658.

Celikoglu, H. B. and Cigizoglu, H. K., 2007. Public transportation trip flow modeling with generalized regression neural networks. Advances in Engineering Software, 38(2), 71–79.

Chan, K.Y., Dillon, T.S., Singh, J., Chang, E., 2012. Neural-network-based models for short-term traffic flow forecasting using a hybrid exponential smoothing and Levenberg–Marquardt algorithm. IEEE Trans. Intell. Trans. Syst. 13 (2), 644–654.

Chen, V.C., Tsui, K.L., Barton, R.R., Allen, J.K., 2003. Ch. 7. a review of design and modeling in computer experiments. Handbook of Statistics 22, 231–261.

Chen, X., Zhang, L., He, X., Xiong, C., Li, Z., 2014. Surrogate-based optimization of expensive-to-evaluate objective for optimal highway toll charges in transportation network. Comput. -Aided Civ. Infrastruct. Eng. 29 (5), 359–381.

Chen, X., Zhu, Z., He, X., Zhang, L., 2015. Surrogate-based optimization for solving a mixed integer network design problem. Transport. Res. Record 2497, 124–134.

Cheng, B., Titterington, D.M., 1994. Neural networks: a review from a statistical perspective. Statistical science 2–30.

Chiu, Y.-C., Korada, P., and Mirchandani, P. B., 2005. Dynamic traffic management for evacuation. In 84th Annual Meeting of the Transportation Research Board, Washington, DC.

Chung, B.D., Cho, H.J., Friesz, T.L., Huang, H., Yao, T., 2014. Sensitivity analysis of user equilibrium flows revisited. Netw. Spatial Econom. 14 (2), 183–207.

Ciuffo, B., Casas, J., Montanino, M., Perarnau, J., Punzo, V., 2013. Gaussian process metamodels for sensitivity analysis of traffic simulation models: case study of AIMSUN mesoscopic model. Trans. Res. Rec. 2390 (1), 87–98.

Cressie, N., 1993. Statistics for Spatial Data: Wiley Series in Probability and Statistics. Wiley-Interscience, New York. 15, 16.

Cressie, N., Wikle, C.K., 2011. Statistics for Spatio-temporal Data. John Wiley & Sons.

Curriero, F.C., 2006. On the use of non-Euclidean distance measures in geostatistics. Mathematical Geology, 38 (8), 907–926.

Dafermos S. C., 1980. Traffic equilibria and variational inequalities. Trunsp. Sci. 14(l), 42-54.

Dafermos S. C., 1982a. The general multimodal network equilibrium problem with elastic demand. Networks 12(1). 57-72.

Dafetmos, S. C., 1982b. Relaxation algorithms for the general asymmetric traffic equilibrium problem. Trump. Sci. 16(2), 23 I-240.

Daganzo, C.F., 1995. The cell transmission model, part II: network traffic. Trans. Res. Part B 29 (2), 79–93.

Evans, S. P., 1976. Derivation and analysis of some models for combining trip distribution and assignment. Transp. Res. 10(l), 37-57.

Fang, H., Horstemeyer, M.F., 2006. Global response approximation with radial basis functions. Eng. Optim. 38 (04), 407–424.

Fang, K.T., Li, R., Sudjianto, A., 2005. Design and Modeling for Computer Experiments. CRC Press.

Fang, K.T., Wang, Y., 1993. Number-theoretic Methods in Statistics (vol. 51). CRC Press.

Faraki, M., Harandi, M. T. and Porikli, F., 2018. Large-scale metric learning: A voyage from shallow to deep," IEEE Trans. Neural Netw. Learn. Syst., vol. 29, no. 9, pp. 4339–4346.

Friedman, J.H., 1991. Multivariate adaptive regression splines. Ann. Stat. 1–67.

Friesz, T. L., 1985. Transportation network equilibrium, design and aggregation: key developments and research opportunities. Transportation Research Part A: General, 19(5-6), 413-427

Friesz, T.L., 2010. Dynamic Optimization and Differential Games, Springer, New York.

Friesz, T.L., Han, K., Szeto, W.Y., 2018. A Review of Dynamic Traffic Assignment Models and Associated Computational Methods. Working paper. June 24, 2018

Friesz, T.L., Bernstein, D., Smith, T., Tobin, R., Wie, B., 1993. A variational inequality formulation of the dynamic network user equilibrium problem. Oper. Res. 41 (1), 80–91.

Friesz, T.L., Han, K., Neto, P.A., Meimand, A., Yao, T., 2013. Dynamic user equilibrium based on a hydrodynamic model. Trans. Res. Part B 47, 102–126.

Friesz, T.L., Kim, T., Kwon, C., Rigdon, M.A., 2011. Approximate dynamic network loading and dual time scale dynamic user equilibrium. Trans. Res. Part B 45 (1), 176–207.

Friesz, T.L., Kwon, C., Mookherjee, R., 2007. A computable theory of dynamic congestion pricing. Trans. Traffic Theory, 1–26.

Gentile, G., Meschini, L., Papola, N., 2007. Spillback congestion in dynamic traffic assignment: a macroscopic flow model with time-varying bottlenecks. Trans. Res. Part B 41, 1114–1138.

Han, K., 2013. An analytical approach to sustainable transportation network design. PhD dissertation.

Han, D., Lo, H.K., 2002. Two new self-adaptive projection methods for variational inequality problems. Comput. Math. Appl. 43, 1529–1537.

Han, K., Eve, G., Friesz, T.L., 2019. Computing Dynamic User Equilibria on Large-Scale Networks with Software Implementation. Networks and Spatial Economics 19:869–902.

Han K., Gayah V.V., 2015a. Continuum signalized junction model for dynamic traffic networks: offset, spillback, and multiple signal phases. Trans. Res. Part B 77:213–239

Han, K., Friesz, T.L., Szeto, W.Y., Liu, H., 2015b. Elastic demand dynamic network user equilibrium: formulation, existence and computation. Trans. Res. Part B 81, 183–209.

Han, K., Friesz, T.L., Yao, T., 2013a. A partial differential equation formulation of Vickrey's bottleneck model, part I: methodology and theoretical analysis. Trans. Res. Part B 49, 55–74.

Han, K., Friesz, T.L., Yao, T., 2013b. A partial differential equation formulation of Vickrey's bottleneck model, part II: numerical analysis and computation. Trans. Res. Part B 49, 75–93.

Han, K., Friesz, T.L., Yao, T., 2013c. Existence of simultaneous route and departure choice dynamic user equilibrium. Trans. Res. Part B 53, 17–30.

Han, K., Piccoli, B., Friesz, T.L., 2016a. Continuity of the path delay operator for dynamic network loading with spillback. Trans. Res. Part B 92, 211–233.

Han, K., Piccoli, B., Szeto, W.Y., 2016b. Continuous-time link-based kinematic wave model: formulation, solution existence, and well-posedness. Transportmetrica B 4 (3), 187–222.

Han, S., 2003. Dynamic traffic modelling and dynamic stochastic user equilibrium assignment for general road networks. Transportation Research Part B: Methodological, 37(3), pp. 225–249.

Horni, A, Nagel, K and Axhausen, K W., 2016. Introducing MATSim. In: Horni, A, Nagel, K and Axhausen, K W. (eds.) The Multi-Agent Transport Simulation MATSim, Pp. 3–8. London: Ubiquity Press. DOI: http://dx.doi.org/10.5334/baw.1. License: CC-BY 4.

Hastie, T., Tibshirani, R., Friedman, J., 2009. The elements of statistical learning, second edition. Springer, NY.

Haykin, S. O., 2008. Neural networks and learning machines, third ed. Pearson-Prentice Hall, NY.

Huang, H. J., Lam, W. H. K., 2002. Modeling and solving the dynamic user equilibrium route and departure time choice problem in network with queues. Trans. Res. Part B 36 (3), 253–273.Huang 2014

Idé, T., Kato, S., 2009. Travel-time prediction using Gaussian process regression: a trajectory-based approach. In: Proceedings of the SIAM International Conference on Data Mining, pp. 1183–1194.

Jain, P., Kulis, B., Davis, J. V. and Dhillon, I. S., 2012. Metric and kernel learning using a linear transformation," J. Mach. Learn. Res., vol. 13, no. 1, pp. 519–547.

Jakobsson, S., Patriksson, M., Rudholm, J., and Wojciechowski, A., 2010. A method for simulation-based optimization using radial basis functions. Optimization and Engineering, 11(4):501–532.

James, G., Witten, D., Hastie, T., Tibshirani, R., 2013. An Introduction to Statistical Learning. Springer, NY.

Jang, W., Ran, B., Choi, K., 2005. A discrete time dynamic flow model and a formulation and solution method for dynamic route choice. Trans. Res. Part B 39 (7), 593–620.

Jeihani, M., 2010. A review of dynamic traffic assignment computer packages. Journal of the Transportation Research Forum (Vol. 46, No. 2).

Jha, M., Moore, K., and Pashaie, B., 2004. Emergency evacuation planning with microscopic traffic simulation. Transportation Research Record: Journal of the Transportation Research Board, (1886): 40–48.

Jones, D. R., Schonlau, M., Welch, W. J., 1998. Efficient global optimization of expensive black-box functions. J. Global Optim. 13 (4), 455–492.

Jia, H., Cheung, Y. M., and Liu, J., 2016. A new distance metric for unsupervised learning of categorical data," IEEE Trans. Neural Netw. Learn. Syst., vol. 27, no. 5, pp. 1065–1079.

Karoonsoontawong, A. and Waller, S. T., 2010. Integrated network capacity expansion and traffic signal optimization problem: robust bi-level dynamic formulation. Networks and Spatial Economics, 10(4): 525–550.

Kleijnen, J.P.C., 2009. Kriging metamodeling in simulation: a review. Eur. J. Oper. Res. 192 (3), 707–716.

Kleijnen, J.P.C., 2015. Design and Analysis of Simulation Experiments, second ed. Springer, NY.

Konnov, I.V., 2003. Application of the proximal point method to non-monotone equilibrium problems. J. Optim. Theory Appl. 119, 317–333.

Koh, A., 2007. Solving transportation bi-level programs with Differential Evolution. 2007 IEEE Congress on Evolutionary Computation, Singapore, pp. 2243-2250.

Kuwahara, M., Akamatsu, T., 1997. Decomposition of the reactive dynamic assignments with queues for a many-to-many origin-destination pattern. Transportation Research B 31(1), 1-10.

Lee, D. H., Nie, Y., and Chen, A., 2003. A conjugate gradient projection algorithm for the traffic assignment problem. Mathematical and computer modelling, 37(7-8), 863-878.

Li, J., Fujiwara, O. and Kawakami, S., 2000. A reactive dynamic user equilibrium model in network with queues. Transportation Research Part B: Methodological, 34(8), pp.605-624.

Li, R., Sudjianto, A., 2005. Analysis of computer experiments using penalized likelihood in Gaussian Kriging models. Technometrics 47 (2), 111–120.

Lim, D., and Lanckriet, G., 2014. Efficient learning of Mahalanobis metrics for ranking. Proc. 31st Int. Conf. Mach. Learn., pp. 1980–1988.

Liu, Y., Lai, X., and Chang, G.L., 2006. Cell-based network optimization model for staged evacuation planning under emergencies. Transportation Research Record: Journal of the Transportation Research Board, (1964):127–135.

Lo, H. K., and Szeto, W. Y., 2002a. A cell-based dynamic traffic assignment model: formulation and properties. Mathematical and computer modelling, 35(7-8), 849-865.

Lo, H. and Szeto, W., 2002b. A cell-based variational inequality formulation of the dynamic user optimal assignment problem. Trans. Res. Part B 36 (5), 421–443.

Long, J.C., Huang, H.J., Gao, Z.Y., Szeto, W.Y., 2013. An intersection-movement-based dynamic user optimal route choice problem. Oper. Res. 61 (5), 1134–1147.

Lophaven, S. N., Nielsen, H. B., Søndergaard, J., 2002. DACE-a matlab Kriging toolbox, version 2.0.

Lønd, A. and Høst, G., 2003. Spatial covariance modelling in a complex coastal domain by multidimensional scaling. Environmetrics, 14 (3), 307–321.

Matheron, G., 1963. Principles of geostatistics. Econ. Geol. 58 (8), 1246–1266.

May, A. D. and Milne, D. S., 2000. Effects of alternative road pricing systems on network performance. Transportation Research Part A: Policy and Practice, 34(6):407–436.

McKay, M.D., Beckman, R.J., Conover, W.J., 1979. Comparison of three methods for selecting values of input variables in the analysis of output from a computer code. Technometrics 21 (2), 239–245.

Merchant, D.K., Nemhauser, G.L., 1978a. A model and an algorithm for the dynamic traffic assignment problem. Transportation Science 12 (3), 183-199.

Merchant, D.K., Nemhauser, G.L., 1978b. Optimality conditions for a dynamic traffic assignment model. Transportation Science 12 (3), 200-207.

Montgomery, D.C., 2008. Design and Analysis of Experiments. John Wiley & Sons.

Mounce, R., 2006. Convergence in a continuous dynamic queuing model for traffic networks. Trans. Res. Part B 40 (9), 779–791.

Mounce, R., Smith, M., 2007. Uniqueness of equilibrium in steady state and dynamic traffic networks. In: Allsop, R.E., H. Bell, M.G., G. Heydecker, B. (Eds.), Transportation and Traffic Theory. Elsevier, pp. 281–299.

Muñoz, J. C. and Laval, J. A., 2006. System optimum dynamic traffic assignment graphical solution method for a congested freeway and one destination. Transportation Research Part B: Methodological, 40(1):1–15.

Murphy, K.P., 2012. Machine Learning, a Probabilistic Perspective. The MIT Press, Cambridge, MA. Nguyen, S., 1984. Estimating origin destination matrices from observed flows. Elsevier Science Publishers, Amsterdam. Transportation Planning Models, pp. 363–380.

Nagel, K. and Flötteröd, G., 2012. Agent-based traffic assignment: Going from trips to behavioural travelers. In Travel Behaviour Research in an Evolving World–Selected papers from the 12th international conference on travel behaviour research (pp. 261-294). International Association for Travel Behaviour Research.

Nguyen, S., 1984. Estimating origin destination matrices from observed flows. Elsevier Science Publishers, Amsterdam. Transportation Planning Models, pp. 363–380.

Nguyen, B., and De Baets, B., 2018. An approach to supervised distance metric learning based on difference of convex functions programming. Pattern Recognit., vol. 81, pp. 562–574.

Nguyen, B., Morell, C., and Baets, B. De., 2018. Distance metric learning for ordinal classification based on triplet constraints. Knowl. -Based Syst., vol. 142, pp. 17–28.

Nguyen, B., Morell, C., and Baets, B. De., 2016. Large-scale distance metric learning for k-nearest neighbors regression," Neurocomputing, vol. 214, pp. 805–814.

Nie, Y.M., Zhang, H.M., 2010. Solving the dynamic user optimal assignment problem considering queue spillback. Netw. Spatial Econom. 10 (1), 49–71.

Osorio, C. and Bierlaire, M., 2011. Mitigating network congestion: analytical models, optimization methods and their applications. In 90th Annual Meeting, number EPFL-TALK-196049.

Park, D., Rilett, L. R., and Han, G., 1999. Spectral basis neural networks for real-time travel time forecasting. Journal of Transportation Engineering, 125(6):515–523.

Peeta, S. and Ziliaskopoulos, A. K., 2001. Foundations of dynamic traffic assignment: The past, the present and the future. Networks and Spatial Economics, 1(3-4), 233–265.

Perakis, G., Roels, G., 2006. An analytical model for traffic delays and the dynamic user equilibrium problem. Oper. Res. 54 (6), 1151–1171.

Queipo, N. V., Haftka, R. T., Shyy, W., Goel, T., Vaidyanathan, R., and Tucker, P. K., 2005. Surrogate-based analysis and optimization. Progress in aerospace sciences, 41(1):1–28.

Regis, R.G., Shoemaker, C.A., 2013. Combining radial basis function surrogates and dynamic coordinate search in high-dimensional expensive black-box optimization. Eng. Optim. 45 (5), 529–555.

Rockafellar, R. T., 1976. Monotone operators and the proximal point algorithm. SIAM journal on control and optimization, 14(5), 877-898.

Sattayhatewa, P. and Ran, B., 2000. Developing a dynamic traffic management model for nuclear power plant evacuation. In 79th annual meeting of the Transportation Research Board, Washington, DC.

Sbayti, H. and Mahmassani, H., 2006. Optimal scheduling of evacuation operations. Transportation Research Record: Journal of the Transportation Research Board, (1964):238–246.

Sacks, J., Welch, W.J., Mitchell, T.J., Wynn, H.P., 1989. Design and analysis of computer experiments. Stat. sci. 409–423.

Santner, T.J., Williams, B.J., Notz, W.I., 2013. The Design and Analysis of Computer Experiments. Springer Science & Business Media.

Smith, M.J., Wisten, M.B., 1995. A continuous day-to-day traffic assignment model and the existence of a continuous dynamic user equilibrium. Ann. Oper. Res. 60 (1), 59–79.

Stein, M.L., 2012. Interpolation of Spatial Data: Some Theory for Kriging. Springer Science & Business Media.

Shen, W., Nie, Y., and Zhang, H., 2007. Dynamic network simplex method for designing emergency evacuation plans. Transportation Research Record: Journal of the Transportation Research Board, (2022), 83–93.

Shen, W. and Zhang, H. M., 2009. On the morning commute problem in a corridor network with multiple bottlenecks: Its system-optimal traffic flow patterns and the realizing tolling scheme. Transportation Research Part B: Methodological, 43(3), 267–284.

Sun, S., Xu, X., 2011. Variational inference for infinite mixtures of Gaussian processes with applications to traffic flow prediction. Intell. Trans. Syst. IEEE Trans. 12 (2), 466–475.

Suwansirikul, C., Friesz, T.L., Tobin, R.L., 1987. Equilibrium decomposed optimization: a heuristic for the continuous equilibrium network design problem. Trans. sci. 21 (4), 254–263.

Szeto, W.Y., 2003. Dynamic traffic assignment: Formulations, properties, and extensions. PhD thesis. The Hong Kong University of Science and Technology, China.

Szeto, W.Y., Lo, H.K., 2004. A cell-based simultaneous route and departure time choice model with elastic demand. Trans. Res. Part B 38, 593–612.

Szeto, W.Y. and Lo, H.K., 2005a. Dynamic Traffic assignment: review and future research directions. Journal of Transportation Systems Engineering and Information Technology, 5(5), pp.85-100.

Szeto, W.Y. and Lo, H.K., 2005b. Properties of dynamic traffic assignment with physical queues. Journal of the Eastern Asia Society for Transportation Studies, 6, pp. 2108-2123.

Tang, B., 2008. Latin hypercube designs. Encyclopedia of Statistics in Quality and Reliability. II.

Ukkusuri, S., Han, L., Doan, K., 2012. Dynamic user equilibrium with a path based cell transmission model for general traffic networks. Trans. Res. Part B 46 (10), 1657–1684.

Wang, G.G., Shan, S., 2007. Review of metamodeling techniques in support of engineering design optimization. J. Mech. Des. 129 (4), 370–380.

Wang, J., Shi, Q., 2013. Short-term traffic speed forecasting hybrid model based on chaos-wavelet analysis-support vector machine theory. Trans. Res. Part C 27, 219–232.

Weinberger, K.Q. and Saul, L. K., 2009. Distance metric learning for large margin nearest neighbor classification. J. Mach. Learn. Res., vol. 10, pp. 207–244.

Wardrop, J., 1952. Some theoretical aspects of road traffic research. In: ICE Proceedings: Part II, Engineering Divisions 1, pp. 325–362.

Wu, L. Hoi, S. C. H., Jin, R., Zhu, J. and Yu, N., 2012. Learning bregman distance functions for semi-supervised clustering. IEEE Trans. Knowl. Data Eng., vol. 24, no. 3, pp. 478–491.

Xie, Y., Zhao, K., Sun, Y., Chen, D., 2010. Gaussian processes for short-term traffic volume forecasting. Trans. Res. Record 2165 (1), 69–78.

Yang, H., Bell, M.G., 2007. Sensitivity analysis of network traffic equilibrium revisited: the corrected approach. In: Mathematics in Transport. Elsevier, Netherlands, pp. 373–395.

Yang, H., Zhang, X., Meng, Q., 2007. Stackelberg games and multiple equilibrium behaviors on networks. Trans. Res. Part B 41 (8), 841–861.

Ye, Q., Szeto, W.Y., Wong, S.C., 2012. Short-term traffic speed forecasting based on data recorded at irregular intervals. IEEE Trans. Intell. Transp. Syst. 13 (4), 1727–1737.

Yin, X., Chen, S., Hu, E., and Zhang, D., 2010. Semi-supervised clustering with metric learning: An adaptive kernel method. Pattern Recognit., vol. 43, no. 4, pp. 1320–1333.

Yperman, I., Logghe, S., Immers, B., 2005. The Link Transmission Model: An Efficient Implementation of the Kinematic Wave Theory in Traffic Networks, Advanced or and AI Methods in Transportation. In: Proc. 10th EWGT Meeting and 16th Mini-EURO Conference. Publishing House of Poznan University of Technology, Poznan, Poland, pp. 122–127.

Zhang, L., He, X., Xiong, C., Zhu, Z., 2014. Bayesian Stochastic Kriging Metamodel for Active Traffic Management of Corridors. In: IIE Annual Conference. Proceedings (p. 1790). Institute of Industrial Engineers-Publisher.

Zhu, D.L., Marcotte, P., 2000. On the existence of solutions to the dynamic user equilibrium problem. Trans. Sci. 34 (4), 402–414.

Zou H., Yue Y., Li, Q, Yeh, A.G.O., 2012. An improved distance metric for the interpolation of link-based traffic data using Kriging: a case study of a large-scale urban road network, International Journal of Geographical Information Science, 26:4, 667-689.

VITA

Wenjing Song

Wenjing Song graduated in 2007 from Tsinghua University with a bachelor's degree in Industrial Engineering, and 2012 from the Chinese University of Hong Kong with a Master's degree in Earth System Science on Dean's List. Wenjing Song has expertise in statistical learning, optimization and mathematical programming of dynamic transportation models. Environmental concerns, challenges of big data and the international wide traffic jam and pollution induced issues are of highest interest of Wenjing Song's research. Wenjing Song has conducted varies studies with researchers internationally and supports growth of interdisciplinary research that concerns welfare of all races, women and children. From 2016-2019, Wenjing Song's work is published as journal articles in a top journal series called Transportation Research. Pushing the boundaries of research, Wenjing Song is a supporter of academic equality.