Doctoral Thesis

Sensitivity analysis in the calibration of microscopic traffic models: From theory to implementation

Author(s):
Ge, Qiao

Publication Date:
2016

Permanent Link:
https://doi.org/10.3929/ethz-a-010859490

Rights / License:
In Copyright - Non-Commercial Use Permitted

This page was generated automatically upon download from the ETH Zurich Research Collection. For more information please consult the Terms of use.
QIAO GE

SENSITIVITY ANALYSIS
IN THE CALIBRATION
OF MICROSCOPIC
TRAFFIC MODELS

FROM THEORY TO IMPLEMENTATION

DISS. ETH NO. 23614
SENSITIVITY ANALYSIS IN THE CALIBRATION OF MICROSCOPIC TRAFFIC MODELS: FROM THEORY TO IMPLEMENTATION

A thesis submitted to attain the degree of

DOCTOR OF SCIENCES of ETH ZURICH

(Dr. sc. ETH Zurich)

presented by

QIAO GE

Master of Science, Technische Universität München, Germany
Master of Science, Tsinghua University, China

born on 24.10.1983

citizen of
China

accepted on the recommendation of

Dr. Monica Menendez
Prof. Dr. Kay W. Axhausen
Prof. Vincenzo Punzo
Prof. Ludovic Leclercq

2016
Foreword

Sensitivity Analysis (SA) is a very relevant topic nowadays, as the simulation of complex systems (e.g., traffic) has become a widely used tool in the scientific and the engineering communities. The work of Mr. Qiao Ge covers multiple aspects associated with this topic, ranging from the development of new qualitative SA methods for screening non-influential independent parameters in high dimensional and computationally expensive models, to the adaptation of state of the art methods for the quantitative SA of dependent parameters.

Given the nature of the work, Mr. Ge draws on tools and knowledge from a number of disciplines, including traffic, statistics, operations research, etc. Some of the contributions show a real departure from methods that are commonly used today. For instance, the sequential approach uses a qualitative SA method developed within the framework of this dissertation (the quasi-OTEE) combined with a quantitative SA method (a Kriging metamodel) to provide solutions comparable to those of the variance-based method at a fraction of the computational cost. Other contributions show the benefits of existing methodologies (e.g., copula-based sampling approach) within the transportation field, especially embedded into the SA.

On behalf of the Traffic Engineering research group at the Swiss Federal Institute of Technology, Zurich, I thank Mr. Ge for his perseverance and dedication to advancing our fundamental knowledge in the field of SA. His work is rather interesting from both a practical and a scientific perspective; as well as useful in many different disciplines. Moreover, some of his ideas and findings could be useful beyond the scope of this dissertation, e.g., to better manage uncertainty of big data, or to model risk within complex systems.

Dr. Monica Menendez
Director of Research Group Traffic Engineering
Institute for Transport Planning and Systems (IVT)
ETH Zurich
Abstract

Microscopic traffic simulations have become indispensable tools for designing, evaluating, and optimizing transportation systems. The reliability of the model results is strictly connected to the quality of the model calibration. A challenge arising in this context concerns the correct selection of the calibration parameters. A model Sensitivity Analysis (SA) should be used with this aim. Unfortunately, a proper SA is hardly performed in the common practice within the transportation community.

This dissertation intends to provide standard and pragmatic SA algorithms for the calibration of traffic simulation models. It starts from a comprehensive review of the most important and widely used global SA methods in literature (Chapter 2). This review aims to gain in-depth understanding of different global SA methods regarding their principles, application conditions, and the possible outcomes. Although general SA practices in the wide scientific research fields are reviewed, the attention is given to those existing applications for traffic simulation models, particularly with respect to model calibration and validation.

Following the review, four new SA approaches are developed and implemented in this dissertation (Chapters 3 to 8). Firstly, the quasi-OTEE approach, which significantly improves the efficiency of the classic Elementary Effects (EE) method, is developed for screening non-influential independent parameters (Chapter 3). Secondly, the sequential SA approach, which combines the quasi-OTEE approach and the Kriging-based approach (Chapter 4), is proposed for the quantitative SA of independent parameters (Chapter 5). Thirdly, based on the development of the copula-based sampling approach (Chapter 6), the extended variance-based SA approach is introduced for the quantitative SA of dependent parameters (Chapter 7). Lastly, a non-parametric approach, i.e., the extended EE approach, is implemented for the parameter screening of models with both independent and dependent parameters (Chapter 8).

Several case studies, including both numerical models and traffic simulation models, are conducted to demonstrate and evaluate the performance of the proposed SA approaches. Results show that all SA approaches developed in this dissertation can achieve satisfactory accuracy and efficiency. Therefore, these approaches can be used in practice to perform proper SA according to the specific requirements of different traffic simulation models, especially those high dimensional and computationally expensive ones. In addition, because all SA approaches developed in this dissertation are general approaches, they can also be applied as practical SA tools for any computationally expensive model in other disciplines from the wider scientific community, where simulations are also commonly used.

Finally, this dissertation proposes a general framework for supporting the application of global SA in the calibration of traffic simulation models (Chapter 9). This framework includes both scientific and pragmatic recommendations with respect to e.g., selecting the proper SA method, setting up the SA procedure, and analyzing the SA results. It is expected that the
findings from this dissertation can help improving the accuracy of general SA practices in traffic simulation, and increasing the efficiency and effectiveness of the model calibration. This, in return, will contribute to improvements in the overall quality of traffic simulations.
Zusammenfassung


Im Anschluss werden vier neue SA-Methoden entwickelt und umgesetzt (Kapitel 3 bis 8). Zunächst wird die quasi-OTEE-Methode, welche die Effizienz der klassischen Elementareffekt-methode (EE-Methode) deutlich erhöht, zur Identifizierung von einflusslosen unabhängigen Parametern (Kapitel 3) entwickelt. Als zweites wird die sequenzielle SA-Methode, welcher die quasi-OTEE-Methode mit der Kriging-Methode kombiniert (Kapitel 4), für die quantitative SA von unabhängigen Parametern (Kapitel 5) vorgestellt. Drittens wird mittels Konzipierung des Kopula-basierten Stichprobeverfahrens (Kapitel 6) die verbesserte varianzbasierte SA-Methode für die quantitative SA von abhängigen Parametern eingeführt. Letztlich wird eine nicht-parametrischer Methode, die verbesserte EE-Methode, für die Parameterüberprüfung bei Modellen mit sowohl unabhängigen als auch abhängigen Parametern aufgeführt (Kapitel 8).


Schließlich stellt die vorliegende Arbeit ein allgemeines Schema vor, um die Anwendung
Acknowledgments

Back to 2010, when I finished my master study at TU München, I was excited that my 20-years student life was finally over, and imagined myself as a true engineer like my parents. Then right before Christmas, when I was searching the internet for working positions, an advertisement of PhD position in Traffic Engineering at ETH Zurich jumped into my sight. I felt that this could be a great opportunity to improve my personal skills and achieve something significant. I sent my application (the only one I’ve ever made for PhD) to ETH, and got a phone interview by Dr. Monica Menendez from the Institute for Transport Planning and Systems (IVT) in February 18th, 2011 (exactly the same day today 6 years ago).

I was very lucky, not only because after a few days I was hired as the second PhD student in Monica’s research team, but also because since that time I got an ABSOLUTELY great supervisor (a.k.a. “the Boss”), who has given tremendous supports to my work and family in Zurich. Therefore, I would like to express my sincere and heartfelt gratitude to Monica at first, for her professional insights and scientific guidance, for the trust and freedom she offered to develop my research, for the always pleasant and open-minded working environment, and the most important, for the inspiring talks that gave me strength to confront every challenge.

I am also very grateful to my co-supervisor at ETH Zurich, Prof. Kay W. Axhausen, and the two external co-examiners, Prof. Vincenzo Punzo (University of Naples Federico II, Italy) and Prof. Ludovic Leclercq (University of Lyon, France), for reviewing this dissertation and for serving on my doctoral examination committee. Their valuable comments and suggestions significantly improved the quality of this dissertation and made it more complete.

Although pursuing a PhD is a long and individual journey, my research would have never reached so far if I walked alone. For this reason, I would like to especially thank Dr. Biagio Ciuffo from European Commission Joint Research Center (ECJRC), for teaching me the basics of Sensitivity Analysis (SA) at the beginning of my PhD, for his involvement and encouragement in many of my publications, and for the opportunities he brought to connect me with experts in the SA research. I also would like to thank Dr. Christian Heimgartner (Roland Müller Küsnacht AG) and the City of Zurich for supporting the use of the Zurich simulation model, which formed a key testing bed for the SA algorithms developed in this dissertation. Moreover, I want to thank Dr. Stefano Tarantola (ECJRC) for inviting me to visit JRC at Ispra, and for the joyful discussions and feedbacks to the results presented in Chapters 7 and 8. And, thanks to Prof. Punzo again, the findings contained in this dissertation benefited a lot from participation in the EU COST Action TU0903 (MULTITUDE).

Furthermore, a big thanks should definitely go to my dear colleagues at the Traffic Engineering Research Group (SVT): Dr. Javier Ortigosa (i.e., the funniest and best office mate), Dr. Jin Cao, Dr. Mireia Roca-Riu, Marco Rothenfluh, Kaidi Yang, Haitao He, and Lukas Ambühl. Thank you all for always being there to support me (especially for helping me...
preparing the doctoral examination) and have lots of funs together in this multicultural team. The friendship among us was the constant source of joy throughout my entire PhD life. I also would like to express my gratitude to the colleagues at IVT, especially to Dr. Kirill Müller and Joerg-Peter Lorch for their assistance on $\LaTeX$ and IT.

Last, but certainly not least, I am deeply indebted to my beloved wife, Bing Chen, our parents, and all friends of us, for their endless love and unconditional support during every moment of my life.

Qiao Ge
Zurich, 18.02.2017
# Contents

## List of Figures  xiii

## List of Tables  xv

## 1 Introduction  1

- 1.1 Background  1
- 1.2 Research Objectives  2
- 1.3 Thesis Outline  3

## 2 Review of Global Sensitivity Analysis Methods  7

- 2.1 Introduction  7
- 2.2 Global Sensitivity Analysis Methods  8
  - 2.2.1 Derivative-based approach  8
  - 2.2.2 Sampling-based approach  9
  - 2.2.3 Regression-based approach  11
  - 2.2.4 Screening approach  12
  - 2.2.5 Variance-based approach  12
  - 2.2.6 Metamodel-based approach  16
  - 2.2.7 Monte Carlo Filtering approach  16
  - 2.2.8 Summary of the reviewed SA approaches  17
- 2.3 Sensitivity Analysis for Traffic Simulation Models  18
- 2.4 Summary of the Chapter  21

## 3 Screening Approach for Model with Independent Parameters  23

- 3.1 Introduction  24
- 3.2 Review of Elementary Effects Method  24
  - 3.2.1 Elementary Effects method  24
  - 3.2.2 Improvement by sampling with trajectories  25
  - 3.2.3 Improvement by sampling with optimized trajectories  27
- 3.3 Quasi-OTEE Method  28
- 3.4 Case Studies  30
  - 3.4.1 Case study 1: sampling efficiency  30
## 8 Screening Approach for Model with Dependent Parameters 107

8.1 Introduction 108

8.2 Methodology 108

8.2.1 Extended EE for dependent parameters 109

8.2.2 Sampling design 110

8.2.3 Computation of the extended EE 111

8.3 Case studies 113

8.3.1 Case study 1: numerical test functions 113

8.3.2 Case study 2: sequential SA with dependent parameters 121

8.4 Summary of the Chapter 126

## 9 Application Framework for Global Sensitivity Analysis 129

9.1 Introduction 129

9.2 Application Framework 130

9.2.1 Defining the objective of the SA 130

9.2.2 Identifying relevant parameters 132

9.2.3 Selecting the SA method 133

9.2.4 Designing the sampling process 135

9.2.5 Evaluating the model 135

9.2.6 Computing the SA results 136

9.2.7 Checking the validity of the SA results 137

9.3 Summary of the Chapter 138

## 10 Conclusions and Outlook 139

10.1 Summary 139

10.2 Recommendations for Future Research 143

## Bibliography 145

## Appendix A Program for Automatic Simulation in Vissim 157

A.1 Introduction 157

A.2 Source Code 157
# List of Figures

<table>
<thead>
<tr>
<th>Figure</th>
<th>Description</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>2.1</td>
<td>Scatter plot for a linear model with two independent inputs</td>
<td>10</td>
</tr>
<tr>
<td>3.1</td>
<td>Example of a random Morris trajectory for a model with 2 parameters</td>
<td>27</td>
</tr>
<tr>
<td>3.2</td>
<td>Layout of the synthetic Aimsun network</td>
<td>33</td>
</tr>
<tr>
<td>3.3</td>
<td>Detection rates of important parameters by the quasi-OTEE SA</td>
<td>34</td>
</tr>
<tr>
<td>3.4</td>
<td>Zurich network in Vissim</td>
<td>35</td>
</tr>
<tr>
<td>3.5</td>
<td>SA results of the Vissim model</td>
<td>38</td>
</tr>
<tr>
<td>4.1</td>
<td>Comparison of the quasi-OTEE and the Kriging-based approach in terms of performance based on the Aimsun simulations</td>
<td>49</td>
</tr>
<tr>
<td>4.2</td>
<td>SA results based on the Vissim simulations</td>
<td>50</td>
</tr>
<tr>
<td>5.1</td>
<td>SA results of the G function</td>
<td>62</td>
</tr>
<tr>
<td>5.2</td>
<td>SA results of the G* function with Coefficient Set-1</td>
<td>63</td>
</tr>
<tr>
<td>5.3</td>
<td>SA results of the function with Coefficient Set-2</td>
<td>64</td>
</tr>
<tr>
<td>5.4</td>
<td>SA results of the K function</td>
<td>65</td>
</tr>
<tr>
<td>5.5</td>
<td>SA results of the Morris function</td>
<td>66</td>
</tr>
<tr>
<td>5.6</td>
<td>Schematic drawing of the Wiedemann-74 model with different driving regimes and thresholds</td>
<td>68</td>
</tr>
<tr>
<td>5.7</td>
<td>Calculation process of the acceleration in the Wiedemann-74 car-following model</td>
<td>69</td>
</tr>
<tr>
<td>5.8</td>
<td>Results of the standard variance-based SA for the Wiedemann-74 model</td>
<td>74</td>
</tr>
<tr>
<td>5.9</td>
<td>Screening results of the sequential SA for the Wiedemann-74 car-following model</td>
<td>76</td>
</tr>
<tr>
<td>5.10</td>
<td>Results of the sequential SA for the Wiedemann-74 model</td>
<td>77</td>
</tr>
<tr>
<td>6.1</td>
<td>Marginal distribution, scatter plot and the SRCC of the five kinematic inputs based on empirical car-following data</td>
<td>86</td>
</tr>
<tr>
<td>6.2</td>
<td>Inverse CDFs of $\Delta x$, $v_f$, $v_l$, $acc_f$, and $acc_l$ obtained from the empirical data</td>
<td>88</td>
</tr>
<tr>
<td>6.3</td>
<td>Marginal distribution, scatter plot and the SRCC of the five kinematic inputs based on 1,024 random samples</td>
<td>90</td>
</tr>
<tr>
<td>6.4</td>
<td>Marginal distribution, scatter plot and the SRCC of the five kinematic inputs based on 10,000 random samples</td>
<td>91</td>
</tr>
<tr>
<td>Figure</td>
<td>Description</td>
<td>Page</td>
</tr>
<tr>
<td>--------</td>
<td>-----------------------------------------------------------------------------</td>
<td>------</td>
</tr>
<tr>
<td>7.1</td>
<td>Estimated independent and full sensitivity indexes for the 17 factors of Wiedemann-74 car-following model (dependent kinematic inputs)</td>
<td>102</td>
</tr>
<tr>
<td>7.2</td>
<td>Estimated independent and full sensitivity indexes for the 17 factors of Wiedemann-74 car-following model (independent kinematic inputs)</td>
<td>105</td>
</tr>
<tr>
<td>8.1</td>
<td>Results of numerical experiment 1 based on the first test scenario</td>
<td>116</td>
</tr>
<tr>
<td>8.2</td>
<td>Results of numerical experiment 1 based on the second test scenario</td>
<td>117</td>
</tr>
<tr>
<td>8.3</td>
<td>Results of numerical experiment 2</td>
<td>118</td>
</tr>
<tr>
<td>8.4</td>
<td>Results of numerical experiment 3</td>
<td>120</td>
</tr>
<tr>
<td>8.5</td>
<td>Screening results for the 17 factors based on the trajectory design</td>
<td>122</td>
</tr>
<tr>
<td>8.6</td>
<td>Screening results for the 17 factors based on the radial design</td>
<td>123</td>
</tr>
<tr>
<td>8.7</td>
<td>Estimated independent and full sensitivity indexes for the 13 factors of Wiedemann-74 car-following model</td>
<td>125</td>
</tr>
<tr>
<td>9.1</td>
<td>Schematic illustration of the application framework for global SA in calibrating traffic simulation models</td>
<td>131</td>
</tr>
</tbody>
</table>
# List of Tables

<table>
<thead>
<tr>
<th>Table</th>
<th>Description</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>2.1</td>
<td>Summary of the reviewed SA approaches</td>
<td>19</td>
</tr>
<tr>
<td>3.1</td>
<td>Comparison of the three trajectory sampling approaches</td>
<td>31</td>
</tr>
<tr>
<td>3.2</td>
<td>Aimsun parameters for the sensitivity analysis</td>
<td>33</td>
</tr>
<tr>
<td>3.3</td>
<td>Vissim parameters for the sensitivity analysis</td>
<td>36</td>
</tr>
<tr>
<td>4.1</td>
<td>Case study settings for comparing Quasi-OTEE and Kriging-based SA</td>
<td>47</td>
</tr>
<tr>
<td>5.1</td>
<td>Values of the 12 coefficients used in computing the $G$ function</td>
<td>56</td>
</tr>
<tr>
<td>5.2</td>
<td>Values of the 20 coefficients used in computing the $G^*$ function</td>
<td>57</td>
</tr>
<tr>
<td>5.3</td>
<td>Analytical formulations for the first order and total sensitivity indexes for the $G$, $G^*$ and $K$ functions</td>
<td>60</td>
</tr>
<tr>
<td>5.4</td>
<td>Number of original model runs for deriving the sensitivity indexes of the test functions</td>
<td>60</td>
</tr>
<tr>
<td>5.5</td>
<td>25 factors for the sensitivity analysis of the Wiedemann-74 model</td>
<td>72</td>
</tr>
<tr>
<td>6.1</td>
<td>Correlation coefficients from the empirical data</td>
<td>87</td>
</tr>
<tr>
<td>6.2</td>
<td>Correlation coefficients from the two experiments</td>
<td>89</td>
</tr>
<tr>
<td>6.3</td>
<td>Comparison of the random samples from the two experiments</td>
<td>92</td>
</tr>
<tr>
<td>7.1</td>
<td>17 factors analyzed in the sensitivity analysis of the Wiedemann-74 model</td>
<td>101</td>
</tr>
<tr>
<td>8.1</td>
<td>Example of trajectory design and radial design</td>
<td>111</td>
</tr>
<tr>
<td>8.2</td>
<td>Number of model runs required by the extended EE method and the extended variance-based SA</td>
<td>115</td>
</tr>
<tr>
<td>8.3</td>
<td>Summary of the reviewed SA approaches</td>
<td>127</td>
</tr>
</tbody>
</table>
Chapter 1

Introduction

1.1 Background

Along with the development of computational techniques, traffic simulation models are more advanced and realistic nowadays. As a result, traffic simulation has grown into a major resource for researchers and practitioners in the field of traffic engineering. The use of commercial traffic simulators (i.e., software) has become widespread, and these programs have indeed become indispensable tools for designing, evaluating, and optimizing transportation systems. The ability to simulate a proposed project for evaluating its potential benefits and costs prior to its implementation allows for significant optimization of the project and other improvements. This, of course, assumes that the model can accurately and efficiently represent the interactions among drivers, vehicles and road facilities.

In order for a traffic simulator to accurately describe reality, it must be supported by a valid traffic model, and it must be properly calibrated. The simulation model is considered valid if it reflects the real world operations in a reasonable way (e.g., how vehicles move in a road, how they change lanes). In other words, the rules within the model must be coherent, and for most of the cases, make physical sense. Calibration, on the other hand, implies that the model parameters (e.g., driving behavior, desired speed) allow the model to recreate the specific network under certain circumstances (i.e., replicate observations, field measurements, and other empirical data). The calibration is vital, yet it can also be rather complex when the traffic model involved is high dimensional (i.e., it contains a large number of parameters) and/or computationally expensive. Moreover, a large number of model parameters are often unobservable and/or really hard to measure in the field.

Due to the limitation of project budgets (e.g., time, manpower), most calibration procedures cannot afford to calibrate all parameters of the simulation model. Thus, calibration is often carried out for only a limited number of parameters. However, there is usually no formal procedure for selecting these parameters, other than choosing the ones that appear to the model user as most likely to have a significant effect on the simulation result (such criteria is often dictated by former experience). As one could imagine, the selection of an incomplete set of parameters for calibration may lead to multiple issues, including but not limited to, model imprecision, and unrealistic values for the calibrated parameters. These problems should not be a surprise, as in traffic models, there exist many interactions between different parameters. For instance, many of the car-following parameters also have impacts on the
lane-changing model. Hence, focusing on the incorrect set might have a cascading effect.

To avoid the potential cascading effects caused by subjectively choosing the incomplete or wrong set of parameters for calibration, Sensitivity Analysis (SA), which has been widely applied in other research disciplines where simulations are also commonly used, is essentially required. The SA explores the relationship between the simulation output and the parameters (Saltelli et al., 2008). A proper SA, including the initial screening of the parameters, can provide both quantitative and qualitative information regarding the effects of different model parameters and their variations on the simulation results. Therefore, a good SA can be very valuable for the subsequent model calibration. Moreover, through SA, the model user is able to identify the most critical parameters, understand those complex internal mechanisms of the model (especially when the model is a black-box), simplify the model, and finally optimize the total efforts required for the calibration (Punzo et al., 2015).

Unfortunately, no previous research suggests a standard SA method for the calibration of microscopic traffic models. The lack of guidelines and relevant experience directly hinders the application of SA in the model calibration (Antoniou et al., 2014). As a matter of fact, several critical questions still remain for the model users:

- Which SA method would be the best option in the calibration of one specific traffic simulation model?
- How does one SA method compare to the others regarding both performance (e.g., accuracy) and computational requirements (e.g., complexity, time and man power required)?
- How to implement such SA methods in an efficient way?

These questions are not trivial, especially because of the complexity of the traffic models themselves, the differences across models, and the variety and application conditions of different SA algorithms. This dissertation is therefore motivated by these issues. It aims at answering, among other things, the questions posted above, and providing both scientific and pragmatic support for the use of SA in the calibration of traffic models.

### 1.2 Research Objectives

The primary objective of this dissertation is to provide standard and pragmatic SA algorithms for the calibration of microscopic traffic simulation models. Specific objectives are listed as follows:

(i) Gain an in-depth understanding of the advantages and drawbacks of different SA methods when applied in the model calibration process.

(ii) Develop efficient algorithms for the SA of traffic simulation models, especially for those high dimensional and computationally expensive models.

(iii) Implement the SA algorithms and test their performance.

(iv) Provide a standard framework including some common guidelines for the application of SA in the calibration of different traffic simulators.
1.3 Thesis Outline

The overall approach adopted in this research and the corresponding findings are presented through 9 chapters in the remainder of this dissertation. An overview of each chapter is given below:

**Chapter 2**: provides a comprehensive review of the most important and widely used global SA methods and their applications. It includes understanding of different SA methods regarding their strengths and weaknesses, properties and requirements for application, as well as the possible outcomes. Although general SA methods in the wide scientific research fields (e.g., environmental engineering, chemistry) are also reviewed in this dissertation, special attention is given to those methods that have been successfully applied in traffic modeling, particularly for model calibration and validation.

**Chapter 3**: presents the Quasi-Optimized Trajectory based Elementary Effects (Quasi-OTEE) method. This method is developed for screening independent parameters of high dimensional and computationally expensive models. Case studies using Aimsun and Vissim models are included to illustrate the application of this method.

**Chapter 4**: compares the Quasi-OTEE method with the Kriging-based SA, which is a recently developed quantitative SA method. Two experimental studies using the same simulation models as in Chapter 3 are presented to compare these two approaches, and better understand their advantages and disadvantages.

**Chapter 5**: proposes a sequential approach for the SA of independent parameters of a high dimensional and computationally expensive model. This approach is a combination of the Quasi-OTEE SA and the Kriging-based SA. The sequential SA is tested with numerical functions and the Wiedemann-74 car-following model (Wiedemann, 1974).

**Chapter 6**: presents a general sampling approach for dependent parameters. It utilizes Gaussian copula to link the marginal distributions of individual parameters with the global distribution and dependence structure of all parameters. A case study that generates dependent random samples for the kinematic parameters of the Wiedemann-74 car-following model is included to demonstrate the sampling process.

**Chapter 7**: presents a quantitative approach for the SA of dependent parameters. The independent and full sensitivity indexes are estimated through Quasi-Monte Carlo (QMC) simulation. In this chapter, the same Wiedemann-74 car-following model from Chapter 5 is used to demonstrate the application of the variance-based SA method for models with both independent and dependent parameters.

**Chapter 8**: introduces a non-parametric screening approach that can be used for the qualitative SA of dependent parameters. It is an extension of the classic Elementary Effects (EE) method. The performance of the proposed method is tested with case studies including both numerical experiments and the Wiedemann-74 car-following model, and the results are cross compared with those obtained via the variance-based SA.

**Chapter 9**: proposes a framework for the application of SA methods in the calibration of microscopic traffic simulation models. Specifically, this framework provides both scientific and pragmatic recommendations with respect to the selection of different
SA methods, design and set-up procedures, possible outcomes, and approaches for analyzing the SA results.

Chapter 10 concludes the findings of this dissertation and discusses open questions for future research.

Many of the chapters in this dissertation are updated versions of papers published/presented in, or submitted to peer reviewed journals and conferences. These are all original work and first-authored by the doctoral candidate. A list of the relevant papers is provided below:

**Papers in Peer Reviewed Journals**


**Papers in Refereed Conference Proceedings**


Non-refereed Papers, Conference Proceedings, and Technical Reports


Working Papers

Ge, Q. and M. Menendez (2016a) Exploring the variance contributions of correlated model parameters: A sampling-based approach and its application in traffic simulation models, manuscript submitted to *Physica A: Statistical Mechanics and its Applications*. 
Chapter 2

Review of Global Sensitivity Analysis Methods

2.1 Introduction

In Saltelli et al. (2004), Sensitivity Analysis (SA) is defined as “the study of how uncertainty in the output of a model (numerical or otherwise) can be apportioned to different sources of uncertainty in the model input”. SA explores the relationship between the variations of model inputs and the corresponding variations of model outputs. In other words, SA provides qualitative and/or quantitative information regarding the effects of model inputs and their variations on the simulation results. Therefore, for any models in e.g., social, economic, engineering, or natural sciences, SA can be used as an essential ingredient of model building and quality assurance. It can be very useful in identifying errors within the model, distinguishing the most important input spaces, establishing priorities for research, etc. (Saltelli et al., 2008).

The simplest and most intuitive SA method is the One-At-a-Time (OAT) method (Saltelli et al., 2008; Punzo and Ciuffo, 2014b). This method computes the partial derivatives analytically or numerically by varying one parameter at a time while keeping all other parameters fixed to a nominal value. As the computation of partial derivatives is only performed at a limited number of input points, it is indeed a local method. One major shortcoming of this local SA is that it is not possible to effectively investigate the sensitivity information in the entire input space other than those default values. Hence, when the model is non-linear, it is very likely the local SA will yield unreliable results. Another problem is because only one model input is allowed to change at a time, the local SA fails to detect the effects caused by input interactions. Therefore, when the model is non-linear and many input interactions exist (i.e., the usual cases for microscopic simulation models), the local SA method is not recommended.

In Saltelli et al. (2006), it is suggested that “a good sensitivity analysis should conduct analyses over the full range of plausible values of key parameters and their interactions, to assess how impacts change in response to changes in key parameters”. This type of SA

\footnote{In the context of general SA practices, the input can include everything that can be changed so that variations of model output can be produced. Hence, it can refer to an input variable, a factor, a model parameter, or even a simulation scenario in different studies.}
is referred to the global SA. The global SA usually involves a Monte Carlo framework to run the model for a large number of times based on random sampled values taken from the distributions of corresponding model inputs (Punzo and Ciuffo, 2014b). In this way, the entire input space and the distributions of all model inputs are considered in the analysis. In addition, the global sampling guarantees that the sensitivity information for any input is obtained without fixing all other inputs as constants, i.e., the interaction effects can be explored by the global SA.

Because of the above advantages, it is more reasonable to employ the global SA method rather than the local SA method for the calibration of traffic simulation models. Thus, the focus of this chapter is on the most widely used global SA methods. Specifically, these methods can be generally categorized as qualitative or quantitative methods depending on the provided sensitivity information. The qualitative SA methods are typically used to qualitatively rank the model inputs based on their influences on the model outputs. On the contrary, the quantitative SA methods are able to measure the variance contributions of model inputs to the total variance of model output in a numerical way. In the following sections both qualitative and quantitative methods are reviewed.

The rest of this chapter is organized as follows. Section 2.2 provides a detailed review of the most commonly used global SA methods, including the algorithms, application conditions, as well as advantages and limitations. Section 2.3 reviews some of the recent SA applications in microscopic traffic modeling. Section 2.4 summarizes the findings of this chapter.

### 2.2 Global Sensitivity Analysis Methods

During the last decades, global SA methods have been the focus of much research in the scientific community. Several inter-disciplinary techniques have been developed and applied in this research area, and there are currently many methods available. This section gives a review of the most important and commonly used global SA approaches. It is worth noticing that these are general approaches, and there are a number of more specific SA algorithms that fall within each of them.

#### 2.2.1 Derivative-based approach

Suppose the model $Y = f(X)$ contains $k$ parameters, i.e., $X = \{X_1, X_2, \cdots, X_k\}$. The sensitivity with respect to one single parameter $X_i$ is measured at a fixed point $X_0$ across all parameters, i.e., $X_0 = \{X_{1,0}, X_{2,0}, \cdots, X_{k,0}\}$, which can be expressed as:

$$S^p_i = \frac{\partial Y}{\partial X_i |_{X_0}}, \quad (2.1)$$

where the superscript “$p$” indicates this expression is obtained from the computation of the partial derivatives at $X_0$.

Many practitioners consider this method as the most natural way to perform the sensitivity analysis, especially when the model can be formulated analytically (Punzo and Ciuffo, 2014a). Accordingly, efficient computer programs are designed to compute system derivatives, which are further used in, e.g., model calibration, model verification, and model simplification.
This approach is very efficient in terms of computer time (Saltelli et al., 2008). However, this approach will not work when the model input space is uncertain (i.e., $X_0$ is unknown), or when the model is nonlinear (Saltelli et al., 2008). The reason is the partial derivative is only computed at the base point rather than the entire input space. Hence, it does not necessarily provide the global sensitivity information of the model. Moreover, this approach is not able to detect the interaction effects (Campolongo et al., 2011), and it is inefficient in terms of analyst time (Saltelli et al., 2008).

Some recent studies such as Kucherenko et al. (2009), Sobol’ and Kucherenko (2009, 2010), and Lamboni et al. (2013) have proposed approaches in which multi-dimensional averaging of the derivatives is used to explore the interaction effects. A particular method, which is also based on the derivative-based approach, is called Sigma-Normalized Derivatives (SND) (Saltelli et al., 2008). The derivatives are computed at multiple base points in the input space. Then the sensitivity index is derived as a normalized partial derivative function:

$$S_i^\sigma = \frac{\partial Y / \sigma_Y}{\partial X_i / \sigma_{X_i}},$$

where $\sigma_Y$ and $\sigma_{X_i}$ stand for the standard deviation of the output $Y$ and the parameter $X_i$, respectively.

SND is recommended for sensitivity analysis by the guidelines of the Intergovernmental Panel for Climate Change (IPCC, 1999, 2000). However, its application is limited when the model is a black-box. In such case, the analytical computation of SND is impossible, while numerical estimation of SND is often very time consuming (Punzo and Ciuffo, 2014a).

### 2.2.2 Sampling-based approach

The sampling-based SA uses Monte Carlo simulation or other approaches such as Latin Hypercube Sampling (LHS, see Saltelli et al., 2008), quasi-random sampling (e.g., Sobol’ sequence, see Sobol’, 1976) to generate random samples. Suppose $N$ random samples (i.e., each sample is a combination of certain values of all parameters based on their marginal distributions) are generated for the $k$-dimensional model $f$. These samples can be described in the following matrix:

$$\tilde{X} = \begin{bmatrix}
  x_{1,1} & x_{2,1} & \cdots & x_{k,1} \\
  x_{1,2} & x_{2,2} & \cdots & x_{k,2} \\
  \vdots & \vdots & \ddots & \vdots \\
  x_{1,N} & x_{2,N} & \cdots & x_{k,N}
\end{bmatrix},$$

where $x_{i,r}$ $(i \in [1,k], r \in [1,N])$ is the $r$-th random sample of parameter $X_i$.

The model is then executed consecutively by taking values from each row of $\tilde{X}$ as the model inputs (Helton et al., 2006). The model output $\tilde{Y} = [y_1, y_2, \cdots, y_N]^T$ with respect to $\tilde{X}$.
is obtained accordingly:

\[
\tilde{Y} = \begin{bmatrix} y_1 \\ y_2 \\ \vdots \\ y_N \end{bmatrix} = \begin{bmatrix} f(x_{1,1}, x_{2,1}, \cdots, x_{k,1}) \\ f(x_{1,2}, x_{2,2}, \cdots, x_{k,2}) \\ \vdots \\ f(x_{1,N}, x_{2,N}, \cdots, x_{k,N}) \end{bmatrix}.
\] (2.4)

The sensitivity information can therefore be obtained by analyzing the relationship between \( \tilde{Y} \) and \( \tilde{X} \). Applications of sampling-based approach can be found in e.g., Helton et al. (2006) and Mara and Joseph (2008). One particular case of the sampling-based approach is the scatter plot analysis.

For each parameter \( X_i \), \( N \) points at coordinates \( \{x_{r,i}, y_{r}\} \), \( r \in [1, N] \) are plotted in a scatter plot. Accordingly, there will be \( k \) scatter plots corresponding to the \( k \) parameters of the model. The shape of the points cloud in the scatter plot represents the sensitivity of the output with respect to \( X_i \), and it can be visually analyzed.

Figure 2.1: Scatter plot for a linear model with two independent inputs. \( Y = X_1 + X_2 \) with \( X_1 \sim N(0, 1), X_2 \sim N(0, 5) \).

\[ Y = X_1 + X_2, \] (2.5)

where \( X_1 \sim N(0, 1), X_2 \sim N(0, 5) \). By generating a size of 1,000 random samples (i.e., \( N = 1,000 \)) for \( X_1 \) and \( X_2 \) each using the normal distribution, the corresponding input-output (i.e., \( X_1-Y \) and \( X_2-Y \)) can be plotted in the scatter plots (Figure 2.1).

In Figure 2.1(a), the cloud of \( X_1 \) is more or less uniformly distributed across different values of \( X_1 \) (i.e., it looks like a circle). On the contrary, in Figure 2.1(b), the cloud of \( X_2 \)
has a much wider dispersion along with different values of \( X_2 \), and a clear linear relationship between \( X_2 \) and \( Y \) can be determined. It obviously shows that parameter \( X_2 \) has much higher impacts on the variation of model output than parameter \( X_1 \).

This method is one of the simplest ways for performing sensitivity analysis. The complexity for analyzing the results is low, and it does not require additional manipulations. However, since the sampling-based approach needs a certain number of samples to conduct the Monte Carlo experiment, the method is not efficient (sometimes not even feasible) for high dimensional models. In those cases, the time requirements are just too high and this approach might become unpractical (Punzo and Ciuffo, 2014a). Another limitation is that this approach does not consider the dependency among the parameters, i.e., the scatter plots do not allow to investigate the sensitivity of the parameters that are correlated.

### 2.2.3 Regression-based approach

Regression models are often used to approximate the original models, especially when the original models are partially known and when they are linear. A linear regression model can be derived as:

\[
Y = b_0 + \sum_{j=1}^{k} b_j X_j + \epsilon,
\]

where \( b_i (i = 0, \cdots, k) \) are regression coefficients, and \( \epsilon \) is the regression error.

The Monte Carlo experiment as mentioned in Section 2.2.2 can be used for estimating the regression coefficients. \( N \) random samples are generated from the marginal distributions of model parameters (i.e., \( \tilde{X} \) in Equation (2.3)), and \( N \) model outputs are produced (i.e., \( \tilde{Y} \) in Equation (2.4)). The regression coefficients \( \hat{b} = [\hat{b}_0, \hat{b}_1, \cdots, \hat{b}_k]^T \) are derived using a least-squares estimation:

\[
\hat{b} = (\tilde{X}_{\text{Reg}}^T \tilde{X}_{\text{Reg}})^{-1} \tilde{X}_{\text{Reg}}^T \tilde{Y},
\]

where

\[
\tilde{X}_{\text{Reg}} = \begin{bmatrix}
1 & x_{1,1} & x_{1,2} & \cdots & x_{1,k} \\
1 & x_{2,1} & x_{2,2} & \cdots & x_{2,k} \\
\vdots & \vdots & \vdots & \ddots & \vdots \\
1 & x_{N,1} & x_{N,2} & \cdots & x_{N,k}
\end{bmatrix} = \begin{bmatrix} \mathbf{1} & \tilde{X} \end{bmatrix}.
\]

Similar as SND in Equation (2.2), the regression coefficients can be normalized as:

\[
\hat{\beta}_i = \frac{\hat{b}_i \sigma_{X_i}}{\sigma_Y},
\]

which can be used as the sensitivity index.

For a linear model, the standardized regression coefficient \( \hat{\beta}_i \) coincides with the SND \( S_i^\sigma \) in Equation (2.2), i.e., \( \hat{\beta}_i = S_i^\sigma \) for \( \forall i \in [1, k] \). However, when the model is non-linear, the regression coefficient is more robust and reliable than SND. The reason is that \( \hat{\beta}_i \) is derived...
using random samples generated from the entire input space, while $S_i^*$ is computed at the midpoint of the marginal distribution of $X_i$ and all other parameters are fixed (Saltelli et al., 2008). Applications of the regression-based approach can be found in e.g., Doksum and Samarov (1995) and Oakley and O’Hagan (2004). Limitations to the regression-based method are imposed by the sample size of the Monte Carlo experiment: the sensitivity index will lack precision when the sample size is small, while a larger sample size will of course take more time for the computation.

2.2.4 Screening approach

The screening approach is a qualitative SA method. It is typically employed to identify non-influential parameters of a model. One commonly used method within this approach is the Elementary Effects (EE) method, developed by Morris (1991). Applications of the screening approach can be found in e.g., Campolongo et al. (2007, 2011), Saltelli et al. (2009), and Wang et al. (2015). The detailed information about the EE method will be described in Chapter 3.

The screening method usually requires a relatively low computational cost for running the model. This feature makes it quite attractive especially for complex models. It can also be used to prune the number of parameters to be considered, before applying a more complicated method such as the variance-based method. One drawback of the screening method is, according to Kucherenko et al. (2009), that it is not able to provide straightforward information regarding the total effects (see Section 2.2.5), and it lacks accuracy in ranking the parameters if compared with the variance-based method (for details see the review in Chapter 3).

2.2.5 Variance-based approach

The variance-based approach was first introduced by Çukier et al. (1973), and Sobol’ (1993) implemented it using the Monte Carlo experiment. Applications of the variance-based approach can be found in e.g., Sobol’ (1993), Saltelli and Sobol’ (1995), Homma and Saltelli (1996), and Saltelli et al. (1999, 2010). The idea of this approach is to use the conditional variance as a measure to represent the influence of the parameter.

Suppose the model $Y = f(X)$ is a square integrable function, and the input space is a $k$-dimensional unit hypercube, i.e., $\mathbb{H}^k = [0,1]^k$ where $\mathbb{H}^k = \{X|0 \leq x_i \leq 1, i \in [1,k]\}$. According to Sobol’ (1993), $f$ can be expanded as:

\[
f(X) = f_0 + \sum_{i=1}^{k} f_i + \sum_{i=1}^{k} \sum_{j=i+1}^{k} f_{ij} + \cdots + f_{1,\ldots,k}, \tag{2.10}
\]

where $f_i = f_i(X_i)$, $f_{ij} = f_{ij}(X_i, X_j)$, and so on for the higher dimensional terms.

The above expansion is known as High Dimensional Model Representation (HDMR)

---

3 For the ease of describing the variance decomposition, the assumption used in the original work of Sobol’ (1993), i.e., the input space is a unit hypercube, is kept here. For real problems the model inputs could have various non-uniform distributions, yet these distributions can always be transformed to the uniform distribution by isoprobabilistic transformations (Lemaire et al., 2010).
(Rabitz, 1989; Saltelli et al., 2008). For a model with only independent inputs, Sobol’(1993) proved that the HDMR expansion is unique if all terms in Equation (2.10) have zero mean, i.e., \( \int_{\mathbb{I}^{k-1}} f_{i_1,i_2,\cdots,i_{k-1}} \, dx_{i_w} = 0 \), where \( 1 \leq i_1 \leq \cdots \leq i_s \leq k \) and \( i_w \in \{ i_1, i_2, \cdots, i_s \} \). This unique decomposition is known as ANalysis Of VAriances (ANOVA) decomposition. Moreover, since the terms in the ANOVA decomposition are pairwise orthogonal (Sobol’, 1993), the decomposition is known as ANOVA decomposition. Moreover, since the terms in the ANOVA decomposition are pairwise orthogonal (Sobol’, 1993), the first three terms in the right side of Equation (2.10) in the context of ANOVA decomposition can be derived as:

\[
\begin{align*}
    f_0 &= E(Y) = \int_{\mathbb{I}^k} f(X) \, dX, \\
    f_i &= E(Y|X_i) - f_0 = \int_{\mathbb{I}^{k-1}} f(X) \, dX_{-i} - f_0, \\
    f_{ij} &= E(Y|X_i,X_j) - f_i - f_j - f_0 = \int_{\mathbb{I}^{k-2}} f(X) \, dX_{-(i,j)} - f_i - f_j - f_0,
\end{align*}
\]

where \( X_{-i} \) includes all model inputs except \( X_i \), and \( X_{-(i,j)} \) includes all model inputs except \( X_i \) and \( X_j \). \( E(\cdot) \) and \( E(\cdot|\cdot) \) stand, respectively, for the expectation and conditional expectation. Other higher dimensional terms in Equation (2.10) can be computed in a similar way.

Accordingly, the variance of model output \( V(Y) \) can be decomposed as:

\[
V(Y) = \sum_{i=1}^{k} V_i + \sum_{i=1}^{k} \sum_{j=i+1}^{k} V_{ij} + \cdots + V_{1,\cdots,k},
\]

where

\[
\begin{align*}
    V_i &= V(f_i) = V[E(Y|X_i)], \\
    V_{ij} &= V(f_{ij}) = V[E(Y|X_i,X_j)] - V_i - V_j,
\end{align*}
\]

and so on for the variances at higher dimensions. Here \( V(\cdot) \) stands for the variance. \( V_i \) is the first order or main variance (i.e., the variance contributed by \( X_i \) alone), and \( V_{ij} \) is the second order variance (i.e., the variance due to the interaction between \( X_i \) and \( X_j \)).

According to Sobol’(2001), the total variance \( V_i^T \) contributed by model input \( X_i \) is:

\[
V_i^T = V_i + \sum_{j \neq i} V_{ij} + \cdots + V_{1,\cdots,k} = E[V(Y|X_{-i})] = V(Y) - V[E(Y|X_{-i})],
\]

where \( V(\cdot|\cdot) \) means the conditional variance. Obviously, \( V_i^T \) contains \( V_i \) and all higher order variances (i.e., the variance caused by input interactions) contributed by \( X_i \).

The analytical formulas for computing \( V(Y) \), \( V_i \) and \( V_i^T \) are shown below:

\[
\begin{align*}
    V(Y) &= \int_{\mathbb{I}^k} f^2(X) \, dX - f_0^2, \\
    V_i &= \int_0^1 \left[ \int_{\mathbb{I}^{k-1}} f(X) \, dX_{-i} \right]^2 dX_i - f_0^2, \\
    V_i^T &= V(Y) - V_{-i} = \int_{\mathbb{I}^k} f^2(X) \, dX - \int_{\mathbb{I}^{k-1}} \left[ \int_0^1 f(X) \, dX_i \right]^2 dX_{-i}.
\end{align*}
\]
Two sensitivity indexes can be computed by normalizing \( V_i \) and \( V_i^T \) with \( V(Y) \):

\[
S_i = \frac{V_i}{V(Y)}, \tag{2.16a}
\]

\[
S_i^T = \frac{V_i^T}{V(Y)}. \tag{2.16b}
\]

\( S_i \) is the **first order sensitivity index**, and \( S_i^T \) is the **total sensitivity index**. For a model with independent inputs, it is obvious that \( S_i \) and \( S_i^T \) should always be between 0 and 1. \( S_i \) should never exceed \( S_i^T \), i.e., \( S_i \leq S_i^T \). The equality only holds when \( X_i \) has no interaction with any other input. According to the above definition, \( S_i \) measures the main variance contribution by \( X_i \). Hence, a high \( S_i \) means \( X_i \) must be an influential input, but a low \( S_i \) does not necessarily mean \( X_i \) is not important as it may have high interaction effects. \( S_i^T \) measures the total variance contribution by \( X_i \), therefore, a low \( S_i^T \) (e.g., \( S_i^T \approx 0 \)) indicates that \( X_i \) is a non-influential input, and it can be fixed without influencing the variance of model output.

Moreover, for a model with independent inputs, the ANOVA decomposition and the two sensitivity indexes mentioned above can be used to determine the model structure (Saltelli et al., 2008; Mara and Tarantola, 2012):

1) if \( \sum_{i=1}^{k} S_i < 1 \) and \( \sum_{i=1}^{k} S_i^T > 1 \), the model is a non-additive model and the interaction effects among model inputs are not trivial; and

2) if \( \sum_{i=1}^{k} S_i = 1 \) and \( \sum_{i=1}^{k} S_i^T = 1 \), the model is a purely additive model.

However, according to Oakley and O’Hagan (2004), the above inferences might be invalid when the model contains dependent inputs (a more detailed discussion is provided in Chapter 7).

The variance-based approach is considered as the best available method today to compute the sensitivity indices based purely on model evaluations (Saltelli et al., 2008). However, the analytical computation of \( S_i \) and \( S_i^T \) using Equations (2.15a), (2.15b) and (2.15c) requires the computation of \( k \)-dimensional integrals in the input space. The corresponding computation process could become very challenging and impractical when \( f \) is a complicated function. Therefore, it is more practical to employ a Monte Carlo approach for estimating the sensitivity indexes.

The MC estimators for \( S_i \) and \( S_i^T \) are given below (Saltelli; 2002; Saltelli et al.; 2010):

\[
\hat{E}(Y) = \frac{1}{N} \sum_{r=1}^{N} f(x_{i,r}, x_{-i,r}), \tag{2.17}
\]

\[
\hat{V}(Y) = \frac{1}{N} \sum_{r=1}^{N} \left[ f(x_{i,r}, x_{-i,r}) \right]^2 - \left[ \hat{E}(Y) \right]^2, \tag{2.18}
\]

\[
\hat{S}_i = \frac{1}{N \cdot \hat{V}(Y)} \sum_{r=1}^{N} \left[ f(x'_{i,r}, x_{-i,r}) - f(x_{i,r}, x_{-i,r}) \right] f(x'_{i,r}, x_{-i,r}), \tag{2.19}
\]

\[
\hat{S}_i^T = \frac{1}{2N \cdot \hat{V}(Y)} \sum_{r=1}^{N} \left[ f(x'_{i,r}, x_{-i,r}) - f(x_{i,r}, x_{-i,r}) \right]^2, \tag{2.20}
\]

where \( N \) is the size of the Monte Carlo experiment, and \( r \) is the index of the random sample (\( r \in [1, N] \)). \( x_{i,r} \) and \( x'_{i,r} \) are two different samples independently generated from the
distribution of $X_i$. In other words, $x_{i,r}$ and $x'_{i,r}$ are independent of each other but they follow the same marginal distribution. Similarly, $x_{-i,r}$ and $x'_{-i,r}$ are two different random samples independently generated from the distribution of $X_{-i}$.

In practice, the quasi-Monte Carlo (QMC) estimator is usually employed as an efficient alternative to the standard MC estimator. The idea behind the QMC estimator is to adopt the quasi-random sequence rather than the pure or pseudo random sequence in the experiment. Although the numbers in a quasi-random sequence share some features of random numbers, they are neither pure random nor pseudo random. However, due to their lower discrepancy, they are able to cover the input space more quickly and equally than pure or pseudo random numbers (Sobol’, 1976). As a result, the QMC estimator has much faster convergence in computing sensitivity indexes than the standard MC estimator (Kucherenko et al., 2011).

There are different types of quasi-random sequences such as Halton, Faure, Niederreiter, and Sobol’ sequences (Niederreiter, 1992). The Sobol’ sequence (Sobol’, 1976) has been proven to outperform many other available quasi-random sequences due to its high efficiency (Jäckel, 2002; Sobol’ et al., 2011). It is therefore widely employed in the QMC estimator for computing the variance-based sensitivity indexes (Saltelli, 2002). First, a matrix containing $N$-by-$2k$ Sobol’ quasi-random numbers is produced:

$$
U = \begin{bmatrix}
    u_{1,1} & \cdots & u_{1,k} & \cdots & u_{1,2k} \\
    u_{2,1} & \cdots & u_{2,k} & \cdots & u_{2,2k} \\
    \vdots & \ddots & \vdots & \ddots & \vdots \\
    u_{N,1} & \cdots & u_{N,k} & \cdots & u_{N,2k}
\end{bmatrix}
$$

(2.21)

Then the $r$-th independent random samples ($r \in [1, N]$) for parameter $X_i$ ($i \in [1, k]$), i.e., $x_{i,r}$ and $x'_{i,r}$, can be obtained as $x_{i,r} = u_{i,r}$, and $x'_{i,r} = u_{k+i,r}$. Note that if the distribution of parameter $X_i$ is not the uniform distribution $\mathcal{U}[0, 1]$, an isoprobabilistic transformation is required to transform $u_{i,r}$ and $u_{k+i,r}$ back to the original distribution of $X_i$.

To estimate the above sensitivity indexes for all $k$ parameters, the model needs to be evaluated $N$ times with the joint input vector, i.e., \{\{x'_{1,r}, x_{-1,r}\}, \ldots, \{x'_{k+r}, x\}_{-1,r}\}, for every parameter $X_i$ (thus $Nk$ runs in total). Furthermore, it needs $N$ evaluations with the input vectors \{x_{1,r}, x_{-1,r}\} and \{x'_{1,r}, x'_{-1,r}\}, respectively (thus another $2N$ runs). As a result, in total there will be $N(k + 2)$ runs of the model. Note that there is no universal rule for the choice of $N$ in the MC experiment. A rule-of-thumb is to start with a small sample size, and later check the numerical stability of the sensitivity indexes by using e.g., bootstrapping technique (Archer et al., 1997). If the results are not sufficiently stable, a new SA with a larger sample size is required.

One limitation of the variance-based approach is the complexity for implementation. In addition, when the model itself is high dimensional and/or computationally expensive, the high computation demand might make this method unfeasible (Campolongo et al., 2011; Kücherenko et al., 2009).

---

4 The discrepancy is a measure that describes the inhomogeneity of the distribution of a random sequence in a unit hypercube (Jäckel, 2002). In general, the lower the discrepancy, the better the space filling property.
2.2.6 Metamodel-based approach

A metamodel is an abstraction of the simulation model. When the simulation model behaves like a black-box, and/or when it has a very high cost to run, the metamodel can be used to approximate and replace the simulation model in the SA (more details are given in Chapter 4). There are several types of metamodels available, and the most commonly used ones include Kriging models, radial basis functions, linear polynomial response surfaces, polynomial chaos expansions, boosting regression trees, and artificial neural networks (Simpson et al., 2001; Iooss and Lemaître, 2015). Since the metamodel itself is usually computationally cheap, the sensitivity indexes can be efficiently estimated based on the metamodel rather than the simulation model.

Applications of metamodels in estimating the Sobol’ sensitivity indexes can be found in e.g., Iooss et al. (2006), Sudret (2008), and Storlie et al. (2009). In these applications, most efforts are spent on developing the metamodel (e.g., mapping all possible interactions, see Saltelli et al., 2010), and calibrating the metamodel. As these efforts are generally dependent on the number of parameters contained in the model (Saltelli et al., 2010), the computational cost can still be huge when the simulation model is high dimensional. In addition, when interactions among the parameters are not negligible, it is also difficult to achieve a perfect estimate of the metamodel. Another drawback is that metamodelling tools such as those developed by Ratto et al. (2007) are not so straightforward to encode as Monte Carlo simulation (Saltelli et al., 2009).

2.2.7 Monte Carlo Filtering approach

The Monte Carlo Filtering (MCF) approach is adopted when the purpose of SA is to check which parameters can produce the model outputs falling within or beyond certain thresholds, rather than which parameters contribute most or least to the variances of the model outputs (see Saltelli et al., 2004, pp. 151-191, for a detailed review).

A Monte Carlo experiment is run to produce realizations of the output with respect to different sample points in the input space (Saltelli et al., 2004). Based on the given thresholds, the output vector \( Y \) can be divided into two complementary subsets: well-behaved realizations \( B \) and misbehaved realizations \( \bar{B} \). Then, a Rationalized Sensitivity Analysis (RSA, see Young, 1999; Spear et al., 1994), which is an MCF procedure, can be applied to determine which parameters are statistically influential ones in terms of leading the model outputs to the well-behaved or misbehaved subset. Specifically, based on the corresponding realizations of the output (i.e., \( B \) and \( \bar{B} \)), the random samples of any parameter \( X_i \) can also be categorized into two subsets, i.e., \((X_i|B)\) and \((X_i|\bar{B})\) with \( b \) and \( \bar{b} \) samples respectively. Then the distributions of \((X_i|B)\) and \((X_i|\bar{B})\) are compared by statistical tests such as the Kolmogorov–Smirnov (KS) two-sample test (Corder and Foreman, 2014):

\[
d_{b,\bar{b}} = \sup \left| F_b(X_i|B) - F_{\bar{b}}(X_i|\bar{B}) \right|, \tag{2.22}
\]

where \( F_b(X_i|B) \) and \( F_{\bar{b}}(X_i|\bar{B}) \) are the empirical cumulative distribution functions for \((X_i|B)\) and \((X_i|\bar{B})\) respectively, and “\( \sup \cdot \)" is the supremum function. As a general rule, the larger \( d_{b,\bar{b}} \) the bigger the difference between the distributions of \((X_i|B)\) and \((X_i|\bar{B})\). If the distributions of \((X_i|B)\) and \((X_i|\bar{B})\) are significantly different, it means that a clearly
recognizable subset of values of $X_i$ from its range will drive the model output to fall into $B$ (or $\overline{B}$ for the opposite case), and hence $X_i$ is an decisive and important parameter of the model. On the other hand, if the difference of the two distributions is not significant, it means that the values of $X_i$ do not necessarily influence the realizations of the model output, i.e., $X_i$ is unimportant in this case.

It is worth mentioning that the MCF described above can introduce extra dependence (in most cases non-linear dependence) to the parameters, even if they are actually independent of each other in the original input space. For example, suppose the model $Y = X_1 + X_2$ have two parameters $X_1$ and $X_2$ from the uniform distribution $\mathcal{U}[0, 1]$. If the filter is set to be $Y > 1$, then the range of $X_2$ is accordingly changed to $[1 - X_1, 1]$, i.e., $X_2$ is dependent on $X_1$. Therefore, sampling strategies corresponding to dependent parameters are required for performing the SA in such context (an efficient sampling approach for dependent parameters will be introduced in Chapter 6).

### 2.2.8 Summary of the reviewed SA approaches

The above review shows that different statistic measures are employed by different SA approaches for deriving the sensitivity information of the model. Consequently, the SA results (e.g., ranking of the parameters by importance) obtained by two different SA approaches for the same model will not necessarily be the same. Therefore, to avoid the confusion in the results given by different SA approaches, the practitioners should clearly define beforehand the specific SA settings according to their own applications, and then choose and apply the SA approaches according to the settings.

The following three settings are commonly used in SA settings (for details see Saltelli et al., 2004), and they are also relevant for the model calibration.

**Factor Prioritization (FP):** the SA is used to identify the most influential parameters which account for the largest portion of the variance of the model output. When these influential parameters are fixed to certain values, the variance of the model output will be greatly reduced. In other words, in the model calibration these influential parameters should be calibrated and fixed with priority.

**Factor Fixing (FF):** the SA is used to identify the least influential parameters which barely have any influence on the variance of the model output. These parameters can then be fixed to any possible value within their ranges of variation without influencing the output of the model. Therefore, the efficiency of model calibration could be significantly improved by taking these parameters out of the parameter set for calibration.

**Factor Mapping (FM):** the SA is used to determine which values of parameters will make the model outputs fall within or beyond certain thresholds. This setting can help defining the boundary conditions of certain calibration parameters in advance, so that invalid model output can be avoid in the calibration process, i.e., the calibration can be optimized by skipping the invalid regions in the input space.

A summary of the above reviewed SA approaches including the specific settings, features, and application conditions is shown in Table 2.1. Note that the conditions for the application of these SA approaches (e.g., model dimension, time spent per run) mentioned in Table 2.1 are obtained based on the experiences of Saltelli et al. (2008), Punzo and Ciuffo (2014b).
and this dissertation. Because different practitioners may have different goals, hardware configurations (e.g., single computer unit, parallel computation), or project budgets (e.g., time, manpower), these indications should not be taken as a universal standard for deciding which SA approach to use.

On the other hand, these indications can be used as general recommendations for selecting one or multiple SA approaches for specific applications. For instance, when the practitioner just wants to identify which parameters are not influential (i.e., FP setting), but does not care much about the exact numerical relationship between the parameters and the model output, then the screening approach can be recommended for the SA. When the computational cost for running the simulation model is not negligible (e.g., more than 10 minutes per run), and when the model contains many parameters, the practitioner may consider to use the metamodeling technique for simplifying the model first, then perform the variance-based SA based on the metamodel. When the model is linear or quasi-linear, it is more reasonable to employ the derivative-based or regression-based approach for the SA due to their low computational demand.

To sum up, the selection of the SA approach is directly related to the specific requirements and objectives of the analysis, thus it can be a quite subjective process. For some challenging cases in which no advanced information of the model itself is available (e.g., black-box), the practitioners may try some computationally cheap SA approaches (e.g., scatter plot, regression, screening) first, and apply those more accurate but also more time-consuming approaches (e.g., variance-based SA) sequentially (an extensive discussion regarding this issue can be found in Chapter 5).

### 2.3 Sensitivity Analysis for Traffic Simulation Models

As mentioned in the previous sections, SA is an important tool for optimizing model calibration. However, despite its importance, a proper SA for traffic models is barely performed in common practice. There are not many examples describing the application of SA for traffic models. As a matter of fact, the survey carried out within the EU COST Action TU0903 MULTITUDE\(^5\) (Punzo, 2011) indicates that because of the lack of relevant information and guidance, 43% of Aimsun (TSS, 2016) users, 31% of Vissim (PTV, 2016) users, and 24% of Paramics (Quadstone Paramics, 2016) users failed to perform the SA in their previous work or applications. In addition, even when SA is used for analyzing the traffic model in some studies (e.g., Jayakrishnan et al., 2001; Schultz and Rilett, 2004), it only plays a trivial role in the whole analysis (Punzo and Ciuffo, 2009).

Among the few examples found in the literature, the OAT-based SA appears to be the most commonly used approach adopted by practitioners. This is partially due to its simplicity and ease of implementation. For example, Lownes and Machemehl (2006) used the OAT method in the calibration of the Vissim model to find the parameters influencing capacity in congested situations. In a research involving the simulation of intersections using Vissim, Mathew and Radhakrishnan (2010) changed each parameter value by a fixed amount (e.g., 10%) while keeping the default value for other parameters, and evaluated the sensitivity of

\(^5\)MULTITUDE: Methods and tools for supporting the Use, caLibration and validaTIon of Traffic simUlation moDEls (MULTITUDE, 2016).
<table>
<thead>
<tr>
<th>Derivative-based SA (SND)</th>
<th>Sampling-based SA (Scatter plot)</th>
<th>Regression-based SA</th>
<th>Screening Approach</th>
<th>Variance-based SA</th>
<th>Metamodel-based SA</th>
<th>Monte Carlo Filtering</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>SA setting</strong></td>
<td>FP</td>
<td>FM</td>
<td>FP</td>
<td>FF</td>
<td>FP, FF</td>
<td>FP, FM</td>
</tr>
<tr>
<td><strong>Type of analysis</strong></td>
<td>Quantitative</td>
<td>Quantitative</td>
<td>Quantitative</td>
<td>Qualitative</td>
<td>Quantitative</td>
<td>Quantitative</td>
</tr>
<tr>
<td><strong>Coping with factors' interactions</strong></td>
<td>No</td>
<td>Yes</td>
<td>No</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
</tr>
<tr>
<td><strong>Coping with nonlinear model</strong></td>
<td>No</td>
<td>Yes</td>
<td>No(^{(a)})</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
</tr>
<tr>
<td><strong>Sampling from</strong></td>
<td>Distributions</td>
<td>Distributions</td>
<td>Distributions</td>
<td>Discrete levels</td>
<td>Distributions</td>
<td>Distributions</td>
</tr>
<tr>
<td><strong>Model dimension</strong></td>
<td>&lt; 100</td>
<td>&lt; 10</td>
<td>&lt; 100(^{(b)})</td>
<td>20 ~ 100(^{(b)})</td>
<td>20 ~ 100</td>
<td>&lt; 20</td>
</tr>
<tr>
<td><strong>Computation time per model run</strong></td>
<td>1min ~ 1h</td>
<td>&lt; 1h</td>
<td>&lt; 1h</td>
<td>&lt; 1min</td>
<td>&lt; 1h</td>
<td>1h</td>
</tr>
<tr>
<td><strong>Total runs of model</strong></td>
<td>–</td>
<td>1000</td>
<td>500 ~ 1000</td>
<td>(n(k + 1))</td>
<td>(N(k + 2))</td>
<td>100 ~ 1000</td>
</tr>
</tbody>
</table>

Source: Saltelli *et al.* (2008, p.273)

\(k\): model dimension, i.e., number of factors.
\(N\): sample size of the Monte Carlo experiment, \(N \approx 500 \sim 1000\).
\(n\): number of trajectories (see Section 3.2.2 for more details), \(n \approx 4 \sim 10\).

FP: factor prioritization; FF: factor fixing; FM: factor mapping.

\(^{(a)}\)When the model is non-linear but monotonic, it can be effectively applied for the ranked transformed version of the model.

\(^{(b)}\)The corresponding approach can work with groups of factors, and hence a higher model dimension is allowed.
the output to each individual change. Kesting and Treiber (2008) employed the same OAT approach for calibrating the Intelligent Driver Model (IDM) and Velocity Difference Model (VDM). Other examples that applied the OAT-based approach for performing SA can be found in Chatterjee et al. (2009) for studying the impact of Vissim car-following parameters on the work zone capacity, in Leclercq et al. (2011) for exploring the parameters influences on the capacity drop in the extended Newell-Daganzo (ND) merge model, and in Sarlas et al. (2013), Deng (2009), and Kryvobokov et al. (2013) for the SA of macroscopic simulation models. However, as already mentioned in previous sections, the OAT-based SA is indeed a local approach, and it does not provide any sensitivity information regarding parameters interactions.

Another popular SA approach adopted by practitioners in the transportation research community is the ANOVA approach. It uses statistical techniques of ANOVA to quantify and link the output variances to the variances of one or more model parameters. Bartin et al. (2006), and Li et al. (2009) used the ANOVA approach based on a two-level full factorial design (for details see Saltelli et al., 2008) to determine the first order effects from the model parameters in Paramics. However, the two-level full factorial design hinders the detection of the interaction effects. Beegala et al. (2005), Ciuffo et al. (2008), and Punzo and Ciuffo (2009) employed the ANOVA approach based on three-level factorial design to find the most important car-following and lane-changing parameters within the Aimsun model. In these studies the second order interaction effects, however, were detected only in Ciuffo et al. (2008) and Punzo and Ciuffo (2009), in which the full factorial design was adopted. Cunto and Saccomanno (2008) and Duong et al. (2010) applied the ANOVA approach based on Plackett-Burman factorial design for screening the parameters during the calibration and validation of Vissim in a safety performance study. Another example of ANOVA with Vissim car-following parameters can be found in Yannes and Lönnès (2010), in which a five-level full factorial design was employed in the experimental design. Applications of ANOVA in macroscopic traffic models can be found in e.g., Schönfelder and Axhausen (2003, 2010) for measuring the significance levels of the impacts of multiple variables on the activity space indicator.

Along with the development of SA techniques and general guidelines (e.g., Antoniou et al., 2014; Daamen et al., 2013) in the last few years, some more advanced global SA approaches were adopted by certain practitioners in traffic modeling in their recent works. The variance-based approach, which is also based on variance decomposition but more efficient than standard ANOVA, has been introduced for the quantitative analysis. Ciuffo et al. (2014) and Punzo et al. (2015) applied the variance-based approach to investigate the most influential parameters of the IDM car-following model, and calibrated the simplified model based on the SA results. Vieira da Rocha et al. (2015) employed the same variance-based approach to investigate the influence of 6 car-following parameters of the Gipps model on the estimation of vehicle emissions. When the simulation model is complicated, the metamodel-based approach has been used for the SA. Ciuffo et al. (2013) and Ciuffo and Punzo (2014) introduced the Kriging metamodel for approximating the Aimsun mesoscopic model with 7 parameters. In Ciuffo et al. (2013) the variance-based SA was performed based on the Kriging metamodel.

It is worth mentioning that besides the application of a single SA approach, some practitioners also adopt the combined version of multiple SA approaches in their calibration work. Park and Qi (2005, 2006) employed both the scatter plot and ANOVA analysis
based on five-level fractional factorial design in the calibration study of the Vissim model. Asamer et al. (2013) combined the OAT-based approach and the EE method to identify the Vissim parameters that are not sensitive to snowy road conditions, and discard these parameters before performing the model calibration. Ciuffo and Azevedo (2014) and Azevedo et al. (2015) proposed a more systematic framework for the application of multiple SA techniques. In Ciuffo and Azevedo (2014), the parameters with similarities were first grouped and analyzed by the variance-based approach through iterations, then the EE method was applied to screen individual parameters from the influential groups. Azevedo et al. (2015) proposed a multi-step global SA which combined variance-based SA, EE method and Kriging metamodel in the calibration of a traffic simulation model for safety analysis.

To sum up, thanks to the progress made within the transportation research community (e.g., the workshops from the research project MULTITUDE, the development of some national guidelines and online tutorials), an increasing number of transportation practitioners are getting aware of the importance of global SA in traffic modeling nowadays. However, SA is still not considered as a common practice, especially when the model is complex (e.g., involves strong non-linearity, contains a large number of parameters, and computationally expensive). As stated in Punzo and Ciuffo (2014b), “many practical difficulties are hidden behind these [SA] steps and the practitioner will easily realize that only experience will make the entire process smoother.” Another important issue is that the dependency among parameters, which exists quite often within microscopic simulation models (see Kim and Mahmassani, 2011), is not addressed in any of the above reviewed SA applications. The main reason is because of the lack of effective SA approaches for dealing with parameter dependency, which in fact is also a challenging problem for the SA research community (see e.g., Kucherenko et al., 2012; Mara and Tarantola, 2012). That being said, applying a SA approach which is dedicated to independent parameters to a model with dependent parameters will bring significant fallacies to the SA results (a more detailed discussion will be given in Chapter 7). Therefore, a practical framework which implements efficient techniques for complex simulation models containing both independent and dependent parameters is essentially required.

2.4 Summary of the Chapter

This chapter presents a review of the global SA approaches which are widely used by practitioners. Compared to the local SA approaches (e.g., the OAT-based approach), the global SA can explore the entire input space of model parameters, and it is able to detect the interaction effects among multiple parameters.

The features and application conditions of the following SA approaches are extensively reviewed and compared: derivative-based approach (i.e., SND), sampling-based approach (i.e., scatter plot), regression-based approach, screening approach, variance-based approach, metamodel-based approach, and Monte Carlo filtering. It is found that different SA approaches apply different statistic measures for deriving the sensitivity information. Thus, the practitioners should choose the proper SA approach according to the specific SA settings (i.e., FP, FF, and FM) and the requirements of the applications.

This chapter also provides a brief review of the SA practices for traffic simulation models. The review shows that although practitioners are getting more aware of the importance of SA
nowadays, global SA is still not commonly used in the calibration and validation of traffic models. This is mainly due to the fact that many of the traffic simulation models are complex (e.g., high dimensional, black-box, computationally expensive), and there is a shortage of efficient SA approaches. As a result many practitioners choose the simplest OAT-based approach for performing the analysis, even though it fails to provide sensitivity information regarding parameter interactions. In addition, it is found that all SA applications in existing traffic simulation literature can only deal with independent parameters. Thus, to solve these issues, a practical SA framework coping with both independent and dependent parameters from complex simulation models is needed. The details of the corresponding SA approaches as well as the application framework will be developed and introduced in the remaining chapters of this dissertation.
Chapter 3

Screening Approach for Model with Independent Parameters

This chapter is an updated version of the papers:


Chapter 3. Screening Approach for Model with Independent Parameters

3.1 Introduction

This chapter presents a practical and efficient screening approach for models with independent parameters. This approach is especially useful for computationally expensive microscopic traffic models. It is developed based on the Elementary Effects (EE) method (Morris, 1991). It is able to screen the most influential parameters of a complex model through computing and comparing their Sensitivity Indexes (SI). With the improved sampling strategy, this approach is much more efficient than the classic EE method. The approach itself is validated through a numerical analysis, and a case study of a small synthetic network in Aimsun. A more detailed case study of the Zurich network in Vissim is employed to illustrate the application. The results demonstrate that the proposed approach is an effective SA tool for the computationally expensive microscopic traffic models, as well as other complex models in the general scientific community.

The rest of this chapter is organized as follows. Section 3.2 presents a brief review of the classic EE Method and other modifications made until now. Section 3.3 explains the methodology for the proposed SA approach. Section 3.4 illustrates three case studies to test and demonstrate the application of the proposed SA. Section 3.5 summarizes the findings of this chapter.

3.2 Review of Elementary Effects Method

3.2.1 Elementary Effects method

The Elementary Effects (EE) method was first introduced by Morris (1991) for screening the influential parameters from a complex model (Saltelli et al., 2008). It has been successfully applied in different fields such as environmental engineering (Campolongo and Saltelli, 1997) and chemistry (Campolongo et al., 2007). The main features of the EE method are described below.

Let \( f \) be a model with \( k \) independent inputs (i.e., \( X = \{X_1, X_2, \ldots, X_k\} \)) which are defined in the \( k \)-dimensional input space \( \Omega^k (\Omega^k \subset \mathbb{R}^k) \). Let \( Y \) be the model output, i.e., \( Y = f(X) \). If only the \( i \)-th input (i.e., \( X_i, i \in [1, k] \)) is varied with a given value (i.e., \( \Delta \)), while all other inputs remain unchanged, the corresponding model output is \( f(X_1, \ldots, X_{i-1}, X_i + \Delta, X_{i+1}, \ldots, X_k) \). Note that \( \{X_1, \ldots, X_{i-1}, X_i + \Delta, X_{i+1}, \ldots, X_k\} \) should still be inside \( \Omega^k \). The elementary effect corresponding to \( X_i \) (i.e., \( EE_i \)) is defined as:

\[
EE_i = \frac{f(X_1, \ldots, X_{i-1}, X_i + \Delta, X_{i+1}, \ldots, X_k) - f(X)}{\Delta}.
\]  

The above definition employs the OAT design, hence it is indeed a local sensitivity measure. To investigate the global sensitivity, the above calculation needs to be repeated using multiple (i.e., \( N \)) samples of \( X \) which are randomly generated in \( \Omega^k \), and each time only \( X_i \) is changed with \( \Delta \) (see the sampling design in Section 3.2.2). Accordingly, \( N \) EEs can be derived for each \( X_i \). In Morris (1991), the mean (i.e., \( \mu_i \)) and the standard deviation (i.e., \( \sigma_i \)) of these \( N \) EEs are used to describe the sensitivity information of \( X_i \). In Campolongo et al. (2007), the absolute mean (i.e., \( \mu^*_i \)) is proposed to replace \( \mu_i \) to reduce the Type II error.
(i.e., considering an important parameter as non-important, see Saltelli et al., 2008). This is particularly important for cases when the model contains many interactive inputs or the model is not monotonic (for more details see Saltelli et al., 2008). The definitions of these three measures are shown below:

\[
\mu_i = \frac{1}{N} \sum_{r=1}^{N} EE_{i,r},
\]

(3.2a)

\[
\sigma_i = \sqrt{\frac{1}{N-1} \sum_{r=1}^{N} (EE_{i,r} - \mu_i)^2},
\]

(3.2b)

\[
\mu^*_i = \frac{1}{N} \sum_{r=1}^{N} |EE_{i,r}|.
\]

(3.2c)

where \(EE_{i,r}\) is the EE of \(X_i\) obtained with the \(r\)-th random sample of \(X\), and \(|·|\) stands for the absolute value. According to Morris (1991) and Campolongo et al. (2007), \(X_i\) is identified as:

a) a non-influential input, if \(\mu^*_i\) is close to zero;

b) an influential input with linear and additive effects but negligible interactions with other inputs, if \(\mu^*_i\) is high but \(\sigma_i\) is low; or

c) an influential input with non-linear effects and/or strong interactions with other inputs, if both \(\mu^*_i\) and \(\sigma_i\) are high.

In addition, when \(\mu\) is low but \(\mu^*_i\) is high, the parameter will have oscillating effects (i.e., both positive and negative effects) depending on the value of other parameters.

The empirical study of Campolongo et al. (2007) showed that using \(\mu^*\) for ranking independent model inputs could have similar results as using the Sobol’ total sensitivity index (see the definition in Chapter 2), but a formal proof of the link between these two measures is still missing (Kucherenko et al., 2009). Some recent studies found that the EE method is less accurate than e.g., the derivative-based GSA for prioritizing the model inputs for non-monotonic models (Sobol’ and Kucherenko, 2009; 2010). However, the same studies (e.g., Kucherenko et al, 2009; Sobol’ and Kucherenko, 2009; 2010) also emphasized that for initial inputs screening, the high accuracy may not be required, and the EE method can still be considered as a compromise between accuracy and efficiency for solving practical SA problems. In fact, the EE method has been successfully applied for screening independent model inputs by many researchers in different disciplines (e.g., Campolongo et al.; 2007; Ruano et al.; 2012; Nossent and Bauwens; 2012; Sin et al.; 2009; Sohier et al.; 2015).

According to Equation (3.1), two model runs are required to compute one EE for one model parameter: first with the initial inputs \(\{X_1, X_2, \cdots, X_k\}\), and then with the varied inputs \(\{X_1, \cdots, X_{i-1}, X_i + \Delta, X_{i+1}, \cdots, X_k\}\). Suppose in a \(k\)-parameter model, \(N\) EEs are needed for computing the global SI of each parameter, then \(2Nk\) runs of the model are required, i.e., the computational cost of the basic EE method is \(2Nk\).

### 3.2.2 Improvement by sampling with trajectories

To overcome the drawbacks of OAT design in terms of a local SA measure (Saltelli and Annoni, 2010), and enhance the computational efficiency, the classic Morris EE method...
can be improved by adopting different sampling designs such as the trajectory design\(^1\) (e.g., Morris, 1991; Campolongo et al., 2007; Ruano et al., 2012), the cell design (e.g., Saltelli et al., 2009), and the radial design (e.g., Saltelli et al., 2010; Campolongo et al., 2011).

The trajectory design is the most commonly adopted sampling design for computing EE (e.g., Campolongo et al., 2007; Ruano et al., 2012; Nosseit and Bauwens, 2012; Sin et al., 2009; Sohier et al., 2015). An empirical study by Campolongo et al. (2011) showed that the trajectory design\(^2\) and the radial design are superior to the cell design in terms of higher accuracy. In addition, when the sample size is high, the trajectory design and the radial design generally yield similar results (for some test functions in Campolongo et al., 2011, the optimized trajectory design even outperforms the radial design with less errors). Thus, due to its accuracy, simplicity and popularity, the trajectory design is employed within the EE method for the SA of independent parameters\(^3\).

Figure 3.1 gives an example of one Morris trajectory (Morris, 1991) for a model with two parameters, i.e., \(X_1\) and \(X_2\). The first point \(P_1\) is randomly picked in the input space with the coordinates \(\{X_0^1, X_0^2\}\). The second point \(P_2\) is generated by randomly increasing or decreasing the value of one parameter by a certain value \(\Delta\) based on \(P_1\). For example, the coordinates of \(P_2\) in this case are \(\{X_1^0 + \Delta, X_2^0\}\). The third point \(P_3\) is generated in the same way but by changing another parameter based on \(P_2\), and its coordinates are hence \(\{X_1^0 + \Delta, X_2^0 + \Delta\}\).

According to the above definition of EE in Equation (3.1), two EEs can be obtained based on these three sampling points:

\[
EE_{X_1} = \frac{f(P_2) - f(P_1)}{\Delta} = \frac{f(X_1^0 + \Delta, X_2^0) - f(X_1^0, X_2^0)}{\Delta},
\]

\[
EE_{X_2} = \frac{f(P_3) - f(P_2)}{\Delta} = \frac{f(X_1^0 + \Delta, X_2^0 + \Delta) - f(X_1^0 + \Delta, X_2^0)}{\Delta}.
\]

This example shows that for a \(k\)-parameter model, a trajectory with \(k+1\) points can provide \(k\) EEs (one per parameter). Therefore, by randomly sampling \(N\) trajectories in the input space, each parameter can get the same amount of EEs (i.e., \(N\)) as before, but only \(N(k + 1)\) model runs are needed.

Finally, it is worth mentioning that for the ease of computation, a common practice is to assume that the input space is a \(k\)-dimensional unit hypercube (i.e., \(\mathbb{R}^k = [0, 1]^k\)) when generating the Morris trajectory. Later, the practitioner needs to use the isoprobabilistic transformations (Lemaire et al., 2010) to transform the coordinates of the sampling points on all trajectories to the original distribution of the corresponding parameter.

---

1. The trajectory design is also known as winding stairs in Chan et al. (2000). The difference is the trajectory design always produces random trajectories separately, while the winding stairs design joins all random trajectories together.
2. The empirical study in Campolongo et al. (2011) actually adopted an improved version of the trajectory design (i.e., optimized trajectory design) proposed by Campolongo et al. (2007). See Section 3.2.3 for more details.
3. The SA becomes much more complex when the model contains dependent parameters. Therefore, the trajectory design and the radial design are both considered in the extended EE approach for screening dependent parameters. More details are provided in Chapter \(8\).
3.2.3 Improvement by sampling with optimized trajectories

Since all Morris trajectories are randomly generated, it is possible that many of them overlap in a certain area of the input space. Using the overlapping trajectories for data sampling may actually result in a poor representation of the input space while many unnecessary model executions are wasted.

To overcome such drawback, an improved sampling design was proposed by Campo-longo et al. (2007). A subset of the most spread trajectories is taken out from a high number of randomly generated Morris trajectories. The trajectories in this subset are called Optimized Trajectories (OT). The concept “spread” is defined based on the Euclidean distance $d_{T_i,T_j}$ between any two trajectories $T_i$ and $T_j$:

$$d_{T_i,T_j} = \begin{cases} 
\frac{1}{k+1} \sum_{p=1}^{k+1} \sum_{q=1}^{k} \sum_{\eta=1}^{k} \left( X_{p,\eta}^q - X_{i,\eta}^q \right)^2, & i \neq j \\
0, & \text{otherwise}
\end{cases}$$

(3.4)

where $k$ is the number of parameters, $X_{p,\eta}^q$ is the $\eta$-th coordinate of the $p$-th point in trajectory $T_i$. For a set (i.e., $S$) of $n$ trajectories (i.e., $S = \{T_1, T_2, \cdots, T_n\}$), the total distance $D_S$ of this set is defined as:

$$D_S = \sqrt{0.5 \times \left( \sum_{i=1}^{n} \sum_{j=1}^{n} d_{T_i,T_j}^2 \right)}.$$  

(3.5)
By enumerating all sets that contain \( n \) trajectories from the original set of \( N \) random Morris trajectories \((n \ll N)\), the set with the maximum \( D_S \) is selected as the OT set, and the number of model runs is accordingly reduced to \( n(k+1) \). The number of Morris trajectories (i.e., \( N \)) is usually chosen between 500 and 1,000 (e.g., Saltelli et al., 2008; Campolongo et al., 2007; Ruano et al., 2012), but there is currently no standard rule for the choice of \( n \). Although some empirical studies such as Campolongo et al. (2007) and Ruano et al. (2011) have shown that 10 to 20 OT are sufficient to provide satisfactory results, Ruano et al. (2012) and Nossent et al. (2013) argued that a high value of \( n \) (e.g., 100) should be considered, especially when there are many nonlinear interactions among the parameters. As a rule-of-thumb, one possible way to find the optimal \( n \) is to increase the number of optimized trajectories gradually, and calculate the corresponding rankings of the parameters based on the SI. The optimal \( n \) is reached once the rankings (especially the rankings of the most influential parameters) converge. It should be noted that although a high value of \( n \) is expected to raise the accuracy of the SA, it may make the initial screening unfeasible when running the simulation is computationally expensive and/or there are many parameters. In practice, a reasonable \( n \) should be adopted by considering both the accuracy and the efficiency of the SA.

The advantage of using OT for sampling is that with the same number of trajectories, they can cover more sampling points than any other non-OT. Hence, it facilitates a better scanning of the input space without increasing the number of model runs (Saltelli et al., 2008). As a result, the same SI can be achieved by sampling with OT, but the number of model executions is greatly reduced. Moreover, because the OT sampling can ensure a good scan of the input space by itself, it is not necessary to use other sampling approaches such as Latin Hypercube Sampling (LHS, for more details see Saltelli et al., 2008) to produce the Morris trajectories at the beginning (Campolongo et al., 2007).

However, when the model is complex and has many inputs, the combinatorial optimization problem of OT selection can make the OT sampling unfeasible: to find the optimized set of \( n \) trajectories out of the \( N \) Morris trajectories, the total number of possible combinations is calculated as \( N!/\left[ n! \cdot (N-n)! \right] \). In some cases the number of combinations could be huge. For example, when \( n = 10, N = 200 \), there are over \( 2.2 \times 10^{16} \) combinations. In a case like this, although running the model might be computationally feasible, it is unfeasible for a normal PC to check all possible combinations to find the OT. In this chapter, the OT sampling approach is further improved to overcome such drawback. The details are given in the next section.

### 3.3 Quasi-OTEE Method

In this dissertation, the quasi-Optimized Trajectories based Elementary Effects (quasi-OTEE) method is developed. This method inherits the same measures of the classic EE method for screening parameters, but overcomes the aforementioned combinatorial optimization problem of the OT sampling, i.e., the quasi-OT sampling significantly improves the efficiency in generating random samples. The algorithm of the quasi-OT sampling is described below.

In order to find the \( n \) OTs, a set (i.e., \( S_N \)) of \( N \) Morris trajectories, i.e., \([T_1, T_2, \cdots, T_N]\), is randomly generated in the input space. According to Equation (3.4), the distance matrix for
all trajectories is defined as:

\[
d_{S_N} = \begin{bmatrix}
0 & d_{T_1,T_2} & \cdots & d_{T_1,T_N} \\
d_{T_2,T_1} & 0 & \cdots & d_{T_2,T_N} \\
\vdots & \vdots & \ddots & \vdots \\
d_{T_N,T_1} & d_{T_N,T_2} & \cdots & 0
\end{bmatrix}.
\] (3.6)

The total distance (i.e., \(D_{S_N}\)) of all trajectories in \(S_N\) is calculated as:

\[
D_{S_N} = \sqrt{0.5 \left( \sum_{i=1}^{N} \sum_{j=1}^{N} d_{T_i,T_j}^2 \right)}.
\] (3.7)

Let \(S_{N-1}(p_1)\) be a subset that contains all trajectories in \(S_N\) except the trajectory \(T_{p_1}\), i.e., \(S_{N-1}(p_1) = S_N \setminus \{T_{p_1}\}, p_1 \in N, N = \{1, 2, \cdots, N\}\). Note that the operation “\(\setminus\)” means the set difference, i.e., for any two sets \(A\) and \(B\), \(A \setminus B = \{x : x \in A \text{ and } x \notin B\}\). The total distance of all trajectories in the subset \(S_{N-1}(p_1)\) is calculated as:

\[
D_{S_{N-1}(p_1)} = \sqrt{D_{S_N}^2 - \sum_{i=1,i \neq p_1}^{N} d_{T_i,T_{p_1}}^2}.
\] (3.8)

By enumerating all \(N\) subsets \(S_{N-1}(p_1)\) from \(S_N\), and computing the corresponding total distance \(D_{S_{N-1}(p_1)}\) of each subset, the subset \(S_{N-1}(s_1) = S_N \setminus \{T_{s_1}\} (s_1 \in N)\) is chosen as the optimal trajectory set in this step, if \(D_{S_{N-1}(s_1)} \geq D_{S_{N-1}(q_1)}\) for \(\forall q_1 \in N \setminus \{s_1\}\).

In the next step, let \(S_{N-2}(p_2)\) be a subset that contains all trajectories in \(S_{N-1}(s_1)\) except the trajectory \(T_{p_2}\), i.e., \(S_{N-2}(p_2) = S_{N-1}(s_1) \setminus \{T_{p_2}\} = S_N \setminus \{T_{s_1}, T_{p_2}\}, p_2 \in N \setminus \{s_1\}\). Accordingly, by enumerating all \(N - 1\) subsets and computing the corresponding total distances, the optimal trajectory set in this step is \(S_{N-2}(s_2) (s_2 \in N \setminus \{s_1\})\), if \(D_{S_{N-2}(s_2)} \geq D_{S_{N-2}(q_2)}\) for \(\forall q_2 \in N \setminus \{s_1, s_2\}\).

Then the same selection process is performed based on \(S_{N-2}(s_2)\), etc. In each step except the first step, the total distance of a trajectory set is computed based on the total distance of its superset, which is derived in the previous step. For example, if the optimal trajectory set \(S_{N-t+1}(s_{t-1})\) is found and its total distance \(D_{S_{N-t+1}(s_{t-1})}\) is known, then in the next step, the total distance of the \(N - t\) trajectories in the subset \(S_{N-t}(p_t)\) is calculated as:

\[
D_{S_{N-t}(p_t)} = \sqrt{D_{S_{N-t+1}(s_{t-1})}^2 - \sum_{i=1,i \neq s_1, \cdots, s_{t-1} \neq p_t}^{N} d_{T_i,T_{p_t}}^2},
\] (3.9)

where \(p_t \in N \setminus \{s_1, s_2, \cdots, s_{t-1}\}, 2 \leq t \leq N - n\). If compared with Equation (3.7), whose time complexity is \(O(N^2)\), Equation (3.9) is much more efficient in computing the total distance. It does not re-compute the summation of trajectory distances that have already been computed in previous steps, which results in a linear time complexity \(O(N)\).

The number of trajectories contained in the optimal trajectory set is decreased by one
after each step. Finally, there is a set $S_n(s_{N-n})$ containing the $n$ most spreading trajectories from $S_{n+1}(s_{N-n-1})$, i.e., the optimal trajectory set obtained in the previous step. As these $n$ trajectories are not necessarily the same ones found by the aforementioned OT approach, they are called herein quasi-Optimized Trajectories (quasi-OT).

In the quasi-OT generation process, only $N$ possible combinations (i.e., subsets) need to be considered when picking $S_{N-1}(s_1)$ from $S_N$, and $N-1$ possible combinations are considered when picking $S_{N-2}(s_2)$ from $S_{N-1}(s_1)$, etc. The total number of combinations considered to get the final optimal set $S_n(s_{N-n})$ is $(N - n + 1) \cdot (N + n)/2$. When $n \ll N$, this number is much smaller than the number of combinations required by the original OT approach (i.e., $N!/[n! \cdot (N - n)!]$). Using the same example as in the last section, the quasi-OT approach requires only 20,055 combinations, much less than the $2.2 \times 10^{16}$ combinations required by the original OT approach.

### 3.4 Case Studies

In this section three case studies are included to validate and demonstrate the efficiency and accuracy of the quasi-OTEE approach. The first case study cross-compares the efficiency of the proposed quasi-OT sampling algorithm with two sampling algorithms, i.e., the original OT sampling by Campolongo et al. (2007) and the sampling approach developed by Ruano et al. (2012). The second case study is used to check the validity of the results obtained by quasi-OTEE. It employs a synthetic network modeled in Aimsun, and compares the quasi-OTEE SA with the variance-based SA in terms of the detection rate of the most influential parameters. The third case study, which is performed based on a real calibration project with the City of Zurich, Switzerland (more details are provided in Ge and Menendez, 2012a), is used to illustrate the application of the proposed quasi-OTEE approach for screening parameters of a high dimensional and computationally expensive traffic simulation model.

#### 3.4.1 Case study 1: sampling efficiency

In the first case study, 500 trajectory sets (i.e., cases) are randomly generated in the input space. Each set contains 20 Morris trajectories (i.e., $N = 20$) for 20 input variables (i.e., $k = 20$). In each trajectory set, 10 trajectories (i.e., $n = 10$) are selected respectively via the original OT approach (i.e., Approach-1) proposed by Campolongo et al. (2007), via the quasi-OT approach (i.e., Approach-2) proposed in this study, and via a recent developed optimized trajectory approach (i.e., Approach-3) proposed by Ruano et al. (2012).

Since Approach-1 uses the exhaustive search algorithm to find the true OT, it often requires an extremely high computational cost. For the ease of cross-comparing the three approaches, the values of $n$ and $N$ are chosen intentionally in this validation test, so that it is computationally feasible for a normal PC (with a 2.80 GHz Intel Core i5 processor) to check all possible combinations with the original OT approach. Different settings of $n$ and $N$ might be considered in other tests when a High Performance Computer (HPC) is available, although similar conclusions are expected. The comparison results are shown in Table 3.1.

Table 3.1 shows that the quasi-OT are very close to the trajectories obtained by the original OT approach. In most cases (88%) the quasi-OT and the true OT are exactly the same.
Table 3.1: Comparison of the three trajectory sampling approaches.

<table>
<thead>
<tr>
<th></th>
<th>Approach-1(^{(a)})</th>
<th>Approach-2(^{(b)})</th>
<th>Approach-3(^{(c)})</th>
</tr>
</thead>
<tbody>
<tr>
<td>Cases with no differences in trajectories compared to Approach-1</td>
<td>–</td>
<td>441 (88%)</td>
<td>15 (3%)</td>
</tr>
<tr>
<td>Cases with a difference of 1 trajectory compared to Approach-1</td>
<td>–</td>
<td>42 (8%)</td>
<td>23 (5%)</td>
</tr>
<tr>
<td>Maximum difference in trajectories compared to Approach-1</td>
<td>–</td>
<td>4 trajectories in 1 case</td>
<td>7 trajectories in 10 cases</td>
</tr>
<tr>
<td>Average total distance obtained by the 10 optimal trajectories</td>
<td>3101.8</td>
<td>3100.7</td>
<td>3074.9</td>
</tr>
<tr>
<td>Total computation time of the 500 cases</td>
<td>15278s</td>
<td>12s</td>
<td>14s</td>
</tr>
</tbody>
</table>

\(^{(a)}\)Original OT approach proposed by Campolongo et al. (2007).
\(^{(b)}\)Quasi-OTEE approach proposed in this research.
\(^{(c)}\)Sampling approach proposed by Ruano et al. (2012).
Moreover, the average total distances are almost the same. Compared with the trajectories obtained by the optimization approach presented in Ruano et al. (2012), although both approaches spend similar time in generating the trajectories, the quasi-OT always have higher dispersion (i.e., the total distance achieved is larger), which means they have less overlaps and hence can provide a better coverage of the input space. This can be explained as below.

In Approach-3, a set of trajectories (1 < t ≤ n) with the highest Euclidean distance (see Equation (3.4)) are first chosen from each row of the distance matrix (Equation (3.6)), and the final optimized trajectories are obtained by adding trajectories to this set through iterations (more details are given in Ruano et al. 2012). However, as only the Euclidian distance but not the total distance (Equation (3.5)) is considered in the first step of Approach-3, it is possible that the t trajectories are actually not the most spread trajectories. As an example, suppose that there are four trajectories (i.e., T_1, T_2, T_3, and T_4), and the Euclidean distances in the first row of the distance matrix are d_{T_1,T_2}, d_{T_1,T_3}, d_{T_1,T_4}. They are ordered as d_{T_1,T_2} > d_{T_1,T_3} > d_{T_1,T_4}. In the first step of Approach-3, the trajectory set {T_1, T_2, T_3} will be chosen because d_{T_1,T_2} and d_{T_1,T_3} are the two biggest elements in the first row of the distance matrix. However, if d_{T_1,T_3} is just slightly larger than D_{T_1,T_4}, but d_{T_1,T_4} ≪ D_{T_1,T_3}, the total distance D_{T_1,T_3,T_4} can be much smaller than the total distance d_{T_1,T_2,T_4} because \sqrt{0.5 * (d_{T_1,T_2}^2 + d_{T_1,T_3}^2 + d_{T_1,T_4}^2)} ≪ \sqrt{0.5 * (d_{T_1,T_2}^2 + d_{T_1,T_3}^2 + d_{T_1,T_4}^2)}. In such case, as the initial trajectory set found by Approach-3 is non-optimal, it is very likely that the final trajectory set obtained would be less spread than the quasi-OT set obtained by Approach-2 (recall that the quasi-OT set is obtained by always discarding the least spread trajectory in each iteration step).

In addition, it is obvious in Table 3.1 that the quasi-OT sampling takes the shortest computational time (i.e., 12s) in this experiment, especially compared to the original OT sampling which took over 1,000 times longer.

This experiment has shown that the sampling results from the quasi-OT are highly consistent with those from the original OT method in Campolongo et al. (2007), and better than those from the approach introduced in Ruano et al. (2012). In addition, the quasi-OT sampling has the highest efficiency (i.e., least computation time) among the three approaches, and this advantage is especially significant if compared with the original OT method.

### 3.4.2 Case study 2: comparison with variance-based SA (Aimsun model)

A small synthetic network (see Figure 3.2) modeled in Aimsun is employed in the second case study.

The SA is performed for 8 parameters of the model: vehicle reaction time (rT), vehicle reaction time at stop (rTS), vehicle length (vL), jam density (jD), give-way time (gWt), maximum acceleration (mA), random seed of the simulation (seed), and the demand factor (OD). For the sake of simplification, the variability of these parameters is not considered during the simulations (intra-period variability). The value assigned to these parameters in different simulations is extracted from uniform distributions with upper and lower bounds reported in Table 3.2. The demand factor is used as a multiplying factor of the entire OD matrix. In this way, it is possible to test the effects of amplifying/reducing the total traffic over the network on different model outputs. In particular, it is assumed that there is a maximum 25% variation of the overall traffic demand.
Figure 3.2: Layout of the synthetic Aimsun network.

![Network Layout](image)

Table 3.2: Aimsun parameters for the sensitivity analysis.

<table>
<thead>
<tr>
<th>#</th>
<th>Parameter</th>
<th>Range</th>
<th>Unit</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Reaction Time</td>
<td>[0.5, 2]</td>
<td>s</td>
</tr>
<tr>
<td>2</td>
<td>Reaction Time Stop</td>
<td>[0.5, 3]</td>
<td>s</td>
</tr>
<tr>
<td>3</td>
<td>Vehicle Length</td>
<td>[4, 8]</td>
<td>m</td>
</tr>
<tr>
<td>4</td>
<td>Jam Density</td>
<td>[140, 200]</td>
<td>veh/km</td>
</tr>
<tr>
<td>5</td>
<td>Give-way Time</td>
<td>[1, 50]</td>
<td>s</td>
</tr>
<tr>
<td>6</td>
<td>Max acceleration</td>
<td>[1, 5]</td>
<td>m/s²</td>
</tr>
<tr>
<td>7</td>
<td>Random Seed</td>
<td>[1, 2.2 x 10⁹]</td>
<td>–</td>
</tr>
<tr>
<td>8</td>
<td>Demand Factor</td>
<td>[0.75, 1.25]</td>
<td>–</td>
</tr>
</tbody>
</table>

Four model outputs are collected from the whole network: *delay, density, traffic flow*, and *travel time*. These outputs are computed both globally on the whole network and on 11 different sections (those having a number attached in Figure 3.2). As a result, it is possible to investigate the role exerted by the different model parameters both locally and globally. Only total outputs at the end of the simulation period (1 hour) are considered, rather than their time series.

As the computational cost for running this model is fairly cheap (i.e., around 30 seconds per simulation), a complete variance-based SA (for details see Chapter 2) is indeed affordable in this experiment. It is carried out with 40,960 model evaluations (i.e., 4,096 x (8 + 2) runs⁴), and the results are considered as reference. Then, 10 tests using the quasi-OTEE are performed. In each test, 50 quasi-OT are chosen from 500 randomly generated Morris trajectories. In total every test with the quasi-OTEE required 450 (i.e., 50 x (8 + 1)) model runs.

⁴The choice of 4.096 as the sample size in this case study is based on bootstrapping estimation of the numerical stability of the sensitivity indexes (see Section 2.2.5).
The total sensitivity indexes (Chapter 2) calculated in the reference simulations with the variance-based approach are used to derive the true rank of a parameter according to its importance (i.e., the most important parameter has the highest total sensitivity index). In the quasi-OTEE analysis, the sensitivity index $\mu^*$ is used to sort the parameters from most important to least important. For instance, the parameter with the highest $\mu^*$ is selected as the most important parameter. The detection rates of the quasi-OTEE approach across different outputs and tests are obtained based on the results of the variance-based approach. A brief explanation of the method used to compute the detection rate is given below.

Let $V$ and $Q$ be two parameters sets obtained with the variance-based approach and the quasi-OTEE approach, respectively. Each set contains $I$ parameters with the highest total sensitivity indexes or $\mu^*$. Then the quasi-OTEE detection rate of the $I$ important parameters is calculated as the cardinality of the intersection of $V$ and $Q$ (i.e., $V \cap Q$) divided by $I$.

Figure 3.3 shows the aggregated results based on all outputs and tests. It is evident from the figure that, the quasi-OTEE approach has a very high precision in identifying the important parameters of the model. In most cases, it can give the same results of the most important parameters as the variance-based approach, but with much less model runs (almost 1% of the total runs used by the variance-based approach). In other words, the quasi-OTEE approach is able to effectively find the whole set of important parameters and decide which parameters to discard.

Figure 3.3: Detection rates of important parameters by the quasi-OTEE SA. Results are aggregated across all outputs and tests.

In summary, the case studies 1 and 2 have shown that:

i) the quasi-OT sampling significantly improves the efficiency of the original OT sampling in Campolongo et al. (2007);

ii) the quasi-OT sampling is more advanced than the sampling approach proposed in Ruano et al. (2012) in terms of better space coverage; and

iii) the quasi-OTEE has very high precision in identifying important parameters, comparable with the variance-based approach but with much less computational cost.

Thus, the quasi-OTEE can be considered as an ideal approach for the preliminary SA of
computationally expensive models, especially when it is unfeasible to apply the variance-based techniques from the beginning.

### 3.4.3 Case study 3: application in real calibration project (Vissim model)

#### 3.4.3.1 Experimental design

The third case study employs a Vissim network of Zurich, Switzerland (see Figure 3.4). This network encompasses the city downtown, with a total area of around 2.6 km$^2$. It is a complex urban layout with narrow streets, hills, mixed transportation modes, a large amount of pedestrians, etc. Due to these complexities, the simulation has a very high computational demand (around 30 minutes for a 1-hour traffic simulation), especially when compared with the studies that simulate individual links, single intersections or a few intersections along a single road (e.g., Park and Schneeberger, 2003; Gomes et al., 2004; Ahmed, 2005; Wu et al., 2005; Park and Qi, 2005; Yu et al., 2006; Miller, 2009).

Figure 3.4: Zurich network in Vissim. Numbers indicate travel time measurement sections.

Since the number of parameters contained in the Vissim model is too large (192 parameters, see PTV, 2012), it is hard to make any SA feasible. Thus, an initial screening is performed first to narrow the set of critical parameters.

The first cut from 192 parameters in the original model down to 148 is based on the study of Zurich inner city’s traffic patterns and characteristics, model intended uses, and available data. For example, parameters related to bicycle lanes are discarded because bicycles are not
included in the model. Then the remaining 148 parameters are classified based on previous studies, common sense, and own experience: 55 parameters are obtained directly from the macroscopic traffic demand model, and another 79 parameters are regarded as not very influential or not highly variable, so these parameters are fixed to their default values in all simulations.

Finally, 14 parameters (see Table 3.3) are chosen as the most critical SA parameters. These parameters are then used for the subsequent analysis with the quasi-OTEE approach. For each simulation, based on the quasi-OT sampling, the 14 parameters took values from uniform distributions within the ranges shown in Table 3.3: All other Vissim parameters are set to their default values according to PTV (2012), or to the values derived from the demand model.

Table 3.3: Vissim parameters for the sensitivity analysis.

<table>
<thead>
<tr>
<th>#</th>
<th>Parameter</th>
<th>Range</th>
<th>Unit</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Average Standstill Distance</td>
<td>[1, 3]</td>
<td>m</td>
</tr>
<tr>
<td>2</td>
<td>Additive Part of Desired Safety Distance</td>
<td>[0, 4]</td>
<td>–</td>
</tr>
<tr>
<td>3</td>
<td>Multiplicative Part of Desired Safety Distance</td>
<td>[1, 5]</td>
<td>–</td>
</tr>
<tr>
<td>4</td>
<td>Maximum Deceleration (Own)</td>
<td>[-6, -2]</td>
<td>m/s²</td>
</tr>
<tr>
<td>5</td>
<td>Accepted Deceleration (Own)</td>
<td>[-1.5, -0.5]</td>
<td>m/s²</td>
</tr>
<tr>
<td>6</td>
<td>-1 m/s² per Distance (Own)</td>
<td>[50, 150]</td>
<td>m</td>
</tr>
<tr>
<td>7</td>
<td>Max. Deceleration for Cooperative Braking</td>
<td>[-5, -1]</td>
<td>m/s²</td>
</tr>
<tr>
<td>8</td>
<td>Accepted Deceleration (Trailing)</td>
<td>[-1.5, -0.5]</td>
<td>m/s²</td>
</tr>
<tr>
<td>9</td>
<td>-1 m/s² per Distance (Trailing)</td>
<td>[50, 150]</td>
<td>m</td>
</tr>
<tr>
<td>10</td>
<td>Minimum Headway</td>
<td>[0.3, 1]</td>
<td>m</td>
</tr>
<tr>
<td>11</td>
<td>Safety Distance Reduction Factor</td>
<td>[0, 1]</td>
<td>–</td>
</tr>
<tr>
<td>12</td>
<td>Maximum Deceleration for Cooperative Braking</td>
<td>[-5, -1]</td>
<td>m/s²</td>
</tr>
<tr>
<td>13</td>
<td>Lane Change Distance</td>
<td>[150, 250]</td>
<td>m</td>
</tr>
<tr>
<td>14</td>
<td>Emergency Stop Distance</td>
<td>[3, 7]</td>
<td>m</td>
</tr>
</tbody>
</table>

The aim of the SA in this case is to identify the influential parameters whose variations had the greatest impact on travel time, so that the calibration of this complex model in terms of the travel time can be feasible. To this end, 20 road sections in the inner city of Zurich (Figure 3.4) are chosen as the travel time measurement sections. The vehicles’ travel time for passing through each road section are measured for a period of one hour after a 15-minutes warm-up period. As the lengths of the 20 travel time measurement sections are not the same, the Travel Time per Meter Travelled (TTMT, which is defined as travel time divided by the length of the corresponding section, and equivalent to the inverse of the speed) are used to aggregate the results measured from different sections. The final sensitivity indexes are accordingly computed based on the TTMT aggregated over all sections.

The simulation resolution (i.e., the number of times the vehicle’s position is calculated within one simulated second) is set to 10 as required by the adaptive signal control program used by the city of Zurich.
3.4.3.2 Computational cost

The quasi-OTEE analysis in this case study used 40 quasi-OTs \((n = 40)\) that are selected from 500 random Morris trajectories. As there are 14 parameters involved in the SA, in total 600 simulations are required in this case study. The total computation time is around 12 days (including 20 minutes for the quasi-OT selection). Below are some rough estimations of the computational costs of different EE approaches described in Section 3.2 (using the same PC as in Section 3.4.1):

- EE method using 200 random samples for each parameter: 116 days
- EE method with 200 Morris trajectories: 62 days
- EE method with 40 true OT: 12 days + \(5 \times 10^{44}\) days for the selection of true OT

Therefore, if considering that the computation time has been greatly reduced by the quasi-OTEE, although it does not guarantee exactly the same random samples as the true OT, the quasi-OTEE is a good compromise between accuracy and efficiency. Notice that it is possible to use a HPC (e.g., 50 cores with parallel computing) to overcome the computation problem described above. However, the cost and availability of the HPC, and the limits from the simulation itself (e.g., software compatibility with the HPC, security considerations in transferring the data) should not be neglected. Furthermore, even if a HPC is available, the quasi-OTEE approach still has its own advantages due to its high efficiency. Compared to the same number of randomly generated Morris trajectories, the quasi-OT will surely have less overlaps and provide a better coverage of the sampling space. Compared to the true OT, due to the high efficiency of the quasi-OTEE, when spending the same computational cost, a much bigger set of the original Morris trajectories (see Section 3.2 and Section 3.4.1) could be considered with the quasi-OT sampling (i.e., \(N_{\text{quasi-OT}} \gg N_{\text{true-OT}}\)). It is expected that using a bigger set of the original Morris trajectories could lead to a more thorough exploration of the sampling space, and further increase the reliability of the results.

3.4.3.3 Results of case study 3

The SI (i.e., \(\mu^*, \mu, \text{and } \sigma\) of EE) are calculated after finishing the 600 simulations. The results for all 14 parameters are plotted in Figure 3.5.

It should be noted that the SI values presented in Figure 3.5 are context dependent (Morris, 1991), and their scale could be totally different in other studies (e.g., Campolongo and Saltelli, 1997; Campolongo et al., 2011). Hence, when analyzing the results, it is suggested that the practitioners should not focus on any particular value/scale of the SI, but the relative differences among them. In other words, for the screening purpose, it is less meaningful to quantitatively interpret the results than to qualitatively compare the SI of different parameters.

For the ease of demonstrating the results, the parameters with similar SI in the \(\mu^* - \sigma\) plot (Figure 3.5(a)) are grouped by using the K-Means Clustering approach (MacQueen et al., 1967). Five clusters are determined accordingly:

- Cluster 1: Parameter 2 \((\mu^* \approx 0.113, \sigma \approx 0.089)\);
- Cluster 2: Parameter 1 \((\mu^* \approx 0.081, \sigma \approx 0.107)\);
- Cluster 3: Parameter 3 \((\mu^* \approx 0.072, \sigma \approx 0.073)\);
- Cluster 4: Parameters 4, 5, 6, 8, 9, 10, 12, 13 and 14 \((\mu^* \approx 0.065, \sigma \approx 0.102)\);
- Cluster 5: Parameters 7 and 11 \((\mu^* \approx 0.051, \sigma \approx 0.068)\).
Parameters 7 and 11 in Cluster 5 have the lowest $\mu^*$ and $\sigma$, therefore they are categorized as the least influential parameters, i.e., they have the least amount of interactions with other parameters and the model output is rarely influenced by their variations.

It is also clear that Parameter 2 has the highest $\mu^*$, hence it is considered as the most influential parameter. Moreover, Parameter 1 has the second highest $\mu^*$ but the highest $\sigma$, which indicates that the variation of this parameter can cause strong non-linearity effects and/or interactions with other parameters. Therefore, Parameter 1 is also considered as one of the most influential parameters.

To examine the sensitivity of the rest parameters, a wedge formed by two lines is plotted in the $\mu$-$\sigma$ plot (Figure 3.5(b)). These two lines are corresponding to $\mu = \pm 2$SEM (Standard Error of the Mean, calculated as $\sigma/\sqrt{n}$ where $n$ is the sample size, in this case 40). According to Morris (1991), the variation of any parameter lying outside the wedge will surely influence the model output, independently of its adopted value, the range of the variation, and the
adopted values of the other parameters. Therefore, Parameter 3 is also one influential parameter, although comparing to Parameters 1 and 2 its variation has fewer impacts on the model output, and it is less correlated with other parameters. As for the parameters belonging to Cluster 4, a comparison between Figures 3.5(a) and 3.5(b) shows that they all have relatively low $\mu$ but high $\mu^*$ and $\sigma$. This indicates that these parameters will have both positive and negative effects depending on the adopted values of other parameters. Therefore, in order to avoid the Type II error, it is reasonable to expand the set of most influential parameters by including Parameters 4, 13 and 14 (i.e., the parameters with the highest $\mu^*$ and $\sigma$ in Cluster 4) in this case study. However, this may require extra efforts in the subsequent analysis (e.g., model calibration).

To sum up, based on the above SA results, the variations of Parameters 1, 2, 3 (from car-following model), 4 (from lane-changing model), and 13 and 14 (from lane model) are prone to have the greatest impact on travel time as found in this case study. It is rather difficult to explain the choice of the six parameters from the physical point of view, since like many other commercial simulators, the non-open source nature of Vissim also hinders the model users to explore its internal calculation process. Yet, this further highlights the value of the SA approach proposed here, through which the complex relationships among parameters in this black-box can be efficiently discovered.

Finally, it is worth mentioning that the above case study is just used to illustrate the application of the proposed quasi-OTEE approach. The influential parameters discussed here may not necessarily be the same ones in other SA studies for Vissim, as different settings of the same model (e.g., changes in network layout) could drive significant changes in the results. In addition, as the classic EE approach also considers the marginal distributions of each input parameter as uncertain factors in the analysis (for details see Saltelli et al., 2008), it is expected that the boundaries, distributions (see Table 3.3), as well as dependence structure (more details will be discussed in Chapter 8) can also have strong impacts on the final SA results.

Nevertheless, this case study has shown that the quasi-OTEE approach is able to deal with the SA of computationally expensive models such as Vissim, and it can give reasonable screening results with relatively cheap computational cost, which is ideal when other SA techniques are too expensive to be applied from the beginning.

### 3.5 Summary of the Chapter

This chapter presents the quasi-OTEE method, which is a screening approach developed based on the classic Elementary Effects (EE) method. This approach can efficiently identify the influential and non-influential parameters in a complex model with independent parameters via computing and qualitatively comparing the Sensitivity Indexes (i.e., $\mu$, $\sigma$ and $\mu^*$). Such feature has been proven by the case studies performed in this chapter. Case studies 1 and 2 have shown that:

- (i) sampling with quasi-OT is much more efficient than with the original OT proposed by Campolongo et al. (2007);
- (ii) the quasi-OT can achieve better coverage of the input space than other OT modifications proposed in the literature (e.g., Ruano et al., 2012); and
(iii) the quasi-OTEE approach has a very high precision in identifying the important parameters.

In the third case study, the SA is performed for the Zurich network modeled in Vissim, which is a complex microscopic traffic model with many parameters. The findings further highlight the advantages of this method. Through applying the quasi-OTEE approach, the most influential parameters in the case study are properly identified, while the computation time is greatly reduced.

To summarize, for a model with independent inputs, the quasi-OTEE approach can be used in finding the important parameters, and deciding which parameters to discard in model calibration. Due to its high efficiency and accuracy, it is ideal for the preliminary SA of computationally expensive models, especially when those quantitative, yet more complex SA techniques (e.g., variance-based approaches) are unfeasible. It can be easily implemented as a stand-alone tool when the purpose of the study is to identify influential and non-influential parameters. It could also be used to set the foundation for other quantitative techniques if a more refined analysis is desired afterwards (e.g., investigate the non-negligible interactions, identify the correct sensitivity ranks of influential parameters). Furthermore, since the quasi-OTEE is a general SA approach, independent of any specific traffic model, it can be applied as a practical and efficient screening tool for any computationally expensive model in the wider scientific community.
Chapter 4

Cross-Comparison of Quasi-OTEE with Kriging-based Approach

This chapter is based on the results presented in:

4.1 Introduction

This chapter presents an exploratory study of two efficient approaches for global SA of computationally expensive traffic simulation models. The first approach is the quasi-OTEE introduced in Chapter 3. It is a screening approach developed based on the classic EE method but with significant improvement of the efficiency. The case studies in Chapter 3 demonstrated that this approach can properly identify the most influential parameters from a computationally expensive model, for which other quantitative SA techniques are not feasible. In particular, it is found that the quasi-OTEE approach yielded similar results as those obtained with the EE method, but only required a small fraction of its computation time.

The second approach is the Kriging-based approach. It is an improved variance-based approach with higher efficiency. It computes the Sobol’ sensitivity indexes (Sobol’, 1993) based on a Kriging approximation of the simulation model. The application of this approach for the SA of traffic simulation model was presented in Ciuffo et al. (2013), in which the DACE (Design and Analysis of Computer Experiments) toolbox (Lophaven et al., 2002a,b) was recursively used to obtain a robust Kriging metamodel. In addition, it is shown in Ciuffo et al. (2013) that with only 128 and 512 model evaluations, the Sobol’ sensitivity index calculated by the Kriging emulator can achieve almost the same value as those calculated using the variance-based approach (see Saltelli et al., 2008), but with almost 40,000 model evaluations.

To better understand the advantages and disadvantages of the quasi-OTEE and the Kriging-based approach, and make proper recommendations to the potential users, a comparison of these two approaches is performed in this chapter. Two different traffic models, i.e., the synthetic network modeled in Aimsun (see Section 3.4.2), and the real network of the city of Zurich, Switzerland, modeled in Vissim (see Section 3.4.3), are used in the case studies.

The rest of this chapter is organized as follows. Section 4.2 briefly reviews the key features of the Kriging-based approach. Section 4.3 describes the experimental design and discusses the results of the case studies. Section 4.4 summarizes the overall findings of this chapter.

4.2 Kriging-based Sensitivity Analysis

The main idea of the Kriging-based SA is to perform the variance-based SA on a Kriging metamodel rather than the simulation model (i.e., the original model), whose sensitivity needs to be assessed but the computational cost is normally expensive. It comprises two steps:
1) estimating the Kriging metamodel; and
2) performing the variance-based SA on the Kriging metamodel.

Since the metamodel itself is a simplified surrogate model of the simulation model, the second step will remain the same as a normal variance-based SA, which has already been discussed in Chapter 2. Therefore this section will only focus on the key features for developing the Kriging metamodel. More details about the Kriging metamodel can be found in e.g., Jooss (2009), Sacks et al. (1989), and Kleijnen (2009).

The Kriging metamodel extends the Kriging principles in geostatistics to the experimental science (see Matheron, 1963) where deterministic simulations and random simulations are involved. Several studies have shown that this surrogate model can become a powerful statistical framework for efficient computing the model response (e.g., Marrel et al., 2008).
Ciuffo and Punzo (2014), also in the case of traffic simulation models (e.g., Ciuffo et al., 2013).

The simplest form of Kriging is Ordinary Kriging (Kleijnen, 2009), in which the following assumption is made:

\[ w(v) = \mu + \delta(v) + \epsilon(v), \]  

where

- \( v = \) vector of the input variables of the simulation model;
- \( w(v) = \) vector of the corresponding model outputs;
- \( \mu = \) the average model output over the whole domain, which is an unknown constant in Ordinary Kriging;
- \( \delta(v) = \) the additive noise with zero mean, which forms a stationary covariance process; and
- \( \epsilon(v) = \) the white noise for random simulations, which is independent of \( \delta(v) \) and has zero mean and constant variance (for deterministic simulations \( \epsilon(v) \) is not included in the above equation).

Let \( y(v) \) be a vector of the predicted outputs of the Kriging metamodel. For Ordinary Kriging, \( y(v) \) can be estimated as:

\[ y(v) = \lambda(v, V)'w(V), \]  

where

- \( w(V) = \) vector of the already simulated outputs of the simulation model based on \( N \) known input points \( V = \{v_1, v_2, \ldots, v_N\} \) (\( V \subset v \)); and
- \( \lambda(v, V) = \) weights matrix, defined by both \( v \) and \( V \).

Unlike the linear regression model (see Section 2.2.3) in which all weights are fixed\(^1\), the weights as defined in \( \lambda(v, V) \) in the Kriging metamodel are related to the distance between \( V \) and \( v \). More specifically, the weight in Kriging decreases when the distance between \( V \) and \( v \) increases. Another difference between the Kriging metamodel and the linear regression model is that when the simulation is deterministic, the Kriging metamodel is an exact interpolation (i.e., the predictions of the metamodel based on known input points \( y(V) \) should always be equal to \( w(V) \)), while the linear regression model does not necessarily have such feature.

The problem for deriving the optimal weights in Equation (4.2) can be formulated as:

\[
\begin{align*}
\text{minimize} & \quad \text{MSE} [y(v)] = \min_{\lambda} \left[ \mathbb{E} (y(v) - w(v))^2 \right], \\
\text{subject to} & \quad \mathbb{E}(y(v)) = \mathbb{E}(w(v)),
\end{align*}
\]  

where MSE stands for Mean Squared Error and \( \mathbb{E}(\cdot) \) is the expectation.

\(^1\)In a linear regression model, the computation of the weights (i.e., the linear regression coefficients in Equation (2.7)) is based on the already simulated input points and the corresponding model outputs. Hence, they will not take the value of any new input point into account. In other words, the weights will be constants once they are derived.
The solution to the above optimization problem, according to Cressie (1993, p122), is:

\[
\lambda_o = \Sigma^{-1} \left[ c + 1 \frac{1 - 1^{\top} \Sigma^{-1} c}{1^{\top} \Sigma^{-1} 1} \right],
\]

where \( 1 \) is the identity vector. \( \Sigma \) is the covariance matrix of the \( N \) known input points \( V \):

\[
\Sigma = \begin{bmatrix}
1 & \rho(v_1, v_2) & \cdots & \rho(v_1, v_N) \\
\rho(v_1, v_2) & 1 & \cdots & \rho(v_2, v_N) \\
\vdots & \vdots & \ddots & \vdots \\
\rho(v_N, v_1) & \rho(v_N, v_2) & \cdots & 1
\end{bmatrix},
\]

and \( \rho(\cdot, \cdot) \) is a user specified correlation function. \( c \) is the covariance vector of \( v \) and \( V \):

\[
c = \begin{bmatrix}
\rho(v, v_1) \\
\rho(v, v_2) \\
\vdots \\
\rho(v, v_N)
\end{bmatrix}.
\]

Combining Equations (4.2) and (4.4), the Kriging prediction at any untried point \( v_0 \) (i.e., \( v_0 \in v \) and \( v_0 \notin V \)) can be computed as:

\[
y(v_0) = \hat{\mu} + c(v_0, V)' \Sigma^{-1} (w(V) - \hat{\mu} 1),
\]

where \( \hat{\mu} = (1^{\top} \Sigma^{-1} 1)^{-1} 1^{\top} \Sigma^{-1} w(V) \).

In addition, the assumption of a stationary covariance process of \( \delta(v) \) indicates that the correlation function only depends on the distance between the inputs. Therefore, for a model with \( k \) input variables, the correlation between two input vectors \( v_i = \{v_{i,1}, v_{i,2}, \cdots, v_{i,k}\} \) and \( v_j = \{v_{j,1}, v_{j,2}, \cdots, v_{j,k}\} \) (\( v_i, v_j \in v \)) is derived as:

\[
\rho(v_i, v_j) = \prod_{g=1}^{k} \rho \left( |v_{i,g} - v_{j,g}| \right),
\]

Different types of correlation functions can be found in several papers (e.g., Sacks et al., 1989; Mitchell and Morris, 1992; Lin et al., 2000). In this study, the Gaussian correlation function, which is a very popular function for deriving Kriging metamodel, is employed. Consequently, Equation (4.8) is defined as:

\[
\rho(v_i, v_j) = \prod_{g=1}^{k} \exp \left( -\theta_g \left| v_{i,g} - v_{j,g} \right|^2 \right),
\]

where \( \theta_g \) is a correlation parameter that determines the importance of the \( g \)-th input variable in the Kriging (i.e., a low \( \theta_g \) means the \( g \)-th input variable is more influential).

Equations (4.4) and (4.7) indicate that the optimal weights \( \lambda_o \) and the corresponding
Kriging prediction depend on the type and parameters of the correlation function \( \rho(\cdot, \cdot) \). Hence, to derive an optimal Kriging metamodel for a given simulation model, it is required to have a proper estimation of the parameters of the correlation function, i.e., the correlation parameters \( \theta_g \) in the Gaussian correlation function (Equation (4.9)). The Maximum Likelihood Estimation (MLE) can be applied for the parameter estimation, however, the constrained maximization required by MLE is usually a hard problem (Martin and Simpson, 2005).

In this study, the DACE toolbox (Lophaven et al., 2002a,b) is used. First, a size of \( N \) random samples of the inputs are generated to compute the output of the simulation model \( w(V) \). Then, a different set of \( N \) random samples of the inputs are used for estimating the weights of the Kriging metamodel \( \lambda_o \) (Equation (4.4)) as well as the prediction of the metamodel \( y(v_0) \) (Equation (4.7)). These \( N \) random samples are also used to produce the outputs of the simulation model, i.e., \( w(v_0) \). Then \( y(v_0) \) and \( w(v_0) \) are cross-compared. If the metamodel prediction is a good match with the output of the simulation model under certain quality criteria, then the metamodel can be adopted to replace the simulation model. Otherwise, the metamodel will be iteratively estimated using new random samples until the quality criteria are met.

It is obvious that the smaller the difference is expected between the metamodel and the simulation model, the more model evaluations are required. Therefore, the selection of the quality criterion, which is dependent on the goal of the simulation experiment, becomes very important, and it directly influences the feasibility of the use of the metamodel. Ciuffo et al. (2013) argued that when the metamodel is developed for the SA of a computationally expensive model, it is not strictly required for a perfect match between the simulation model and the metamodel. The experiments in Ciuffo et al. (2013) have shown that the estimation of the sensitivity indexes even with an imperfect metamodel can still provide satisfactory results. On the other hand, it is essential that the metamodel should mimic the behavior of the simulation model by correctly reflecting the input-output relationship. It is worth mentioning that the accuracy of Kriging metamodels to reproduce the input-output relationship of a complex simulation model needs to be verified case by case. The reason is different simulation models could have different properties for which the aforementioned Gaussian process does not fit universally.

Once the metamodel is well estimated, the variance-based SA can be performed based on the metamodel rather than the simulation model with \( N(k + 2) \) runs. As each run of the metamodel requires a very small amount of time, a large scale Monte Carlo experiment (i.e., \( N \)) can be affordable. Note that since the estimation of the metamodel will usually take much more time than running it, the total number of runs of the simulation model required for developing the metamodel is the computational cost of the Kriging-based approach in this study.

4.3 Case Studies

4.3.1 Experimental design

In order to evaluate the performance of the quasi-OTEE and Kriging-based SA, and to make the comparison independent of the simulator used so that convincing results could be achieved,
two experiments are conducted in this study. The quasi-OTEE and the Kriging-based approach are directly compared in each experiment.

### 4.3.1.1 Case study 1: Aimsun model

The first experiment employs the same Aimsun model presented in Section 3.4.2. The layout of the simulated network can be seen in Figure 3.2. The model has 8 parameters, i.e., vehicle reaction time ($rT$), vehicle reaction time at stop ($rTs$), vehicle length ($vL$), jam density ($jD$), give-way time ($gWt$), maximum acceleration ($mA$), random seed of the simulation ($seed$), and the demand factor ($OD$). The lower and upper bounds of corresponding input variables, which are assumed to have uniform distributions, can be seen in Table 3.2. Four simulation outputs of the model are considered in this study, i.e., traffic flow, density, travel time, and time delay.

As in the case study in Section 3.4.2, the quasi-OTEE is conducted using 50 quasi-OTs selected from 500 random Morris trajectories. Accordingly, the computational cost for performing quasi-OTEE analysis is 450 in this case.

In order to develop the Kriging model, the Sobol’ quasi-random sequence (Sobol’, 1976) are used to generate the random samples of the inputs. The Sobol’ sequence has the lowest discrepancy when the sample size is a power of two. Hence, in this case study the Kriging metamodel is gradually estimated over 128, 256, 512 and 1024 model evaluations. The test is carried out over 64 further model evaluations for validating the prediction of the metamodel. The quality criteria is arbitrarily set to be that the metamodel is acceptable when the root mean square percentage error over the 64 model evaluations is smaller than 5%. This criteria is achieved with 512 runs of the simulation model, i.e., the computational cost for the Kriging-based SA in this case study is 512. Moreover, in order to check the stability of Kriging-based sensitivity indexes, the 90% confidence intervals are numerically computed by using the bootstrapping technique (Efron and Tibshirani, 1993). More details are given in Section 9.2.6 and Archer et al. (1997) for computing the Bootstrap Confidence Interval (BCI).

To cross-compare the quasi-OTEE SA and the Kriging-based SA, 40,960 runs of the simulation model are used to perform the variance-based SA, and the results are considered as reference. The results from Kriging-based SA (512 simulation runs) and the quasi-OTEE SA (450 runs) are compared with the reference ones.

### 4.3.1.2 Case study 2: Vissim model

The second case uses the Zurich network modeled in Vissim (more details are given in Section 3.4.3). Figure 3.4 shows the layout of the Vissim model as well as the travel time measurement sections. In this study 14 parameters of the model are considered. They are selected from 192 Vissim parameters and are listed below:

a) Car-following model: Average Standstill Distance ($\text{avSD}$), Additive Part of Desired Safety Distance ($\text{aDSD}$), and Multiplicative Part of Desired Safety Distance ($\text{mDSD}$);

b) Lane-changing model: Maximum Deceleration ($mD$), Accepted Deceleration ($aD$), and $-1 \text{ m/s}^2$ per Distance ($\text{Dis}$) for the leading vehicle, Maximum Deceleration ($mDt$), Accepted Deceleration ($aDt$), and $-1 \text{ m/s}^2$ per Distance ($\text{DisT}$) for the trailing vehicle, Minimum Headway ($\text{minH}$), Safety Distance Reduction Factor ($sD$), and Maximum Deceleration for Cooperative Braking ($mDcB$); and
c) Lane model: Lane Change Distance \((lCd)\), and Emergency Stop Distance \((eSd)\).

All these 14 parameters are assumed to have uniform distributions within the range reported in Table 3.3. Only one output, Travel Time per Meter Travelled (TTMT) aggregated over all measurement sections (see Section 3.4.3, is considered in the SA in the second case study.

The quasi-OTEE SA employs 40 quasi-OT for the input sampling, and hence 600 model runs are required to calculate the corresponding elementary effects, i.e., the computational cost required by quasi-OTEE is 600.

The aforementioned Sobol’ quasi-random sequence is also used here for making random samples and developing the Kriging metamodel. The Kriging metamodel is achieved based on the results of 1,024 runs of the Vissim model. It means that the computational cost for the Kriging-based SA in this case is 1,024. This also confirms that the complexity of the model itself is a key factor for determining the number of input combinations required to obtain a satisfactory Kriging emulator.

Due to the high computational cost of running the model, the variance-based approach is not considered to produce reference results in this experiment. Instead, the results of the two approaches are compared directly in terms of number of important parameters identified, and the ranks of the important inputs.

A summary of the settings of the above two case studies is presented in Table 4.1:

<table>
<thead>
<tr>
<th>Case 1</th>
<th>Case 2</th>
</tr>
</thead>
<tbody>
<tr>
<td>Model</td>
<td>Aimsun</td>
</tr>
<tr>
<td>Number of Parameters</td>
<td>8</td>
</tr>
<tr>
<td>Number of Outputs</td>
<td>4</td>
</tr>
<tr>
<td>Computational Cost for Variance-based SA</td>
<td>40’960</td>
</tr>
<tr>
<td>Computational Cost for Quasi-OTEE</td>
<td>450</td>
</tr>
<tr>
<td>Computational Cost for Kriging-based SA</td>
<td>512</td>
</tr>
</tbody>
</table>

### 4.3.2 Results of case study 1: Aimsun model

The reference simulations took 40,960 model runs to compute the total sensitivity index \(S^T\) of each input variable. The variance-based approach is used to provide the sensitivity information of each parameter according to its importance. Similar calculations of \(S^T\) are used in the Kriging-based analysis as well, but they are computed based on the Kriging model with many fewer model runs (i.e., 512 runs for developing the Kriging model). After a careful examination of the results, in both analyses, a threshold value of 0.15 in the total sensitivity index (15% of the total output variance) is used to distinguish the important and non-important parameters, i.e., a parameter is considered as important if its \(S^T\) is higher than 0.15.

In the quasi-OTEE analysis, the sensitivity index \(\mu^*\) derived from 450 model runs is used to provide the sensitivity information of each parameter. In particular, a threshold of 40% of
the maximum $\mu^*$ is used to distinguish the important and non-important parameters, i.e., a parameter is considered to be important if its $\mu^*$ is not less than 40% of the maximum $\mu^*$ in each test.

In this experiment, the performance of the quasi-OTEE and the Kriging-based approach are compared by referencing the results to those from the variance-based analysis, i.e., the accuracy in determining the important and non-important parameters, and their ranks. As the two approaches use different metrics in the SA (i.e., $S_T$ and $\mu^*$), and since there is no formal proof of any direct connection between $S_T$ and $\mu^*$ (Kucherenko et al., 2009), a quantitative comparison of the values of $S_T$ and $\mu^*$ is not necessarily required in this case study. The results are shown in Figure 4.1.

Figures 4.1(a) and 4.1(b) illustrate the accuracy of the two approaches in finding the important and non-important parameters by different sections and outputs. In particular, the percentage of correct assessments (i.e., considering an important parameter from the variance-based SA as important in one of the two approaches and vice-versa), the percentage of Type I error (i.e., considering a non-important parameter as important), and the percentage of Type II error are accordingly evaluated.

Figure 4.1(c) shows the detection rate of the ranks of the most important parameters using the two approaches. The data presented in this figure is aggregated across all types of outputs and sections plus the network. The ranks of the parameters are determined based on $S_T$ for the analysis using the variance-based approach and the Kriging-based approach, and based on $\mu^*$ for the analysis using the quasi-OTEE approach. It should be noted that according to Sobol' and Kucherenko (2009, 2010), ranking based on $\mu^*$ can be different from that based on $S_T$. Here for the ease of showing the results of the comparison, it is assumed that the same ranking rule can still be applied to both $S_T$ and $\mu^*$, i.e., the higher the $S_T$ or $\mu^*$ is, the more important the parameter will be. The detection rate is calculated by comparing the rank of each parameter given by the quasi-OTEE or Kriging-based approach, to the rank given by the variance-based approach.

From the three charts of Figure 4.1, the following conclusions can be draw:

1. Overall, there is almost no variance in performance across types of outputs, but there are some variances in performance across sections. Nevertheless, these variances in performance do not substantially change the main conclusions.

2. The quasi-OTEE always yields fewer Type II errors but more Type I errors than the Kriging-based approach. This indicates that, compared to the Kriging-based approach, the quasi-OTEE has a higher precision in finding the whole set of important parameters. Hence, the quasi-OTEE is better for the initial parameter screening since it is less likely to discard the important parameters at the beginning. As a result, the cascading errors in the subsequent model calibration are minimized or even completely avoided.

3. Although the quasi-OTEE has a higher detection rate of all important parameters, the Kriging-based approach performs better in ranking the important parameters as well as determining the most important ones. Therefore, the Kriging-based approach is preferable for refining the screening results and providing the correct sensitivity ranks of the selected most important parameters.
4.3. Case Studies

Figure 4.1: Comparison of the quasi-OTEE and the Kriging-based approach in terms of performance based on the Aimsun simulations.

(a) Accuracy by sections. The results are aggregated across all outputs.

(b) Accuracy by outputs. The results are aggregated across all sections plus the global network.

(c) Detection rate of the ranks of the most important parameters. The results are aggregated across all outputs and all sections plus network.

4.3.3 Results of case study 2: Vissim model

In the second case study using the Vissim simulations, the SA results obtained by the quasi-OTEE (with 600 simulations) and the Kriging-based approach (with 1,024 simulations) are compared directly. The SA results are shown in Figure 4.2.

It is worth mentioning that the quasi-OTEE and Kriging-based approaches adopt different
Figure 4.2: SA results based on the Vissim simulations. The results are aggregated based on the TTMT for all 20 measurement sections.

(a) SA results from the quasi-OTEE based on the sensitivity index $\mu^*$ of the EE for the 14 parameters. The dash line represents 40% of the maximum $\mu^*$.

(b) SA results from the Kriging-based approach. The bars represent the total sensitivity indexes and the first order sensitivity indexes for the 14 parameters. The dash line represents 15% of the total output variance.

metrics (i.e., $\mu^*$ and $S^T/S$, respectively) to derive the results. $\mu^*$ and $S^T/S$ are calculated differently, and they have different scales. Therefore, when cross comparing the results between Figures 4.2(a) and 4.2(b), the practitioner should focus on the relative differences
rather than the absolute differences among different parameters.

Figure 4.2(a) shows the bar chart of the \( \mu^* \) of the EE for all 14 parameters based on the aggregated TTMT for all 20 intersections. Same as the first experiment, a threshold of 40\% of the maximum \( \mu^* \) is used to distinguish the important and non-important parameters. The \( \mu^* \) of the parameters Additive Part of Desired Safety Distance (\( aDSD \)), Average Standstill Distance (\( aVSD \)), Multiplicative Part of Desired Safety Distance (\( mDSD \)), Lane Change Distance (\( lCd \)), and Maximum Deceleration of the Own Vehicle (\( mD \)) are all above the threshold. Therefore, they can be considered as the important parameters according to the quasi-OTEE SA.

Figure 4.2(b) illustrates the bar chart of the SA results obtained by the Kriging-based approach. It shows the total sensitivity indexes \( S^T \) (black bars) and the first order sensitivity indexes \( S \) (white bars) for the 14 parameters. The results are based on the aggregated TTMT for all 20 travel time measurement sections. The threshold value of 15\% of the total output variance (i.e., 0.15 in the \( S^T \)) is also employed here to identify the important parameters. In Figure 4.2(b), the \( S^T \) of the 3 parameters \( aVSD \), \( aDSD \), and \( mDSD \) from the car-following model are all above 0.15. In addition, these three parameters have both the maximum \( S^T \) and the maximum \( S \). Hence, they are considered as the most important parameters according to the Kriging-based approach.

A cross-comparison between Figures 4.2(a) and 4.2(b) shows that both approaches can successfully capture the 3 most important parameters (i.e., \( aVSD \), \( aDSD \), and \( mDSD \)) in this experiment. Notice also that the quasi-OTEE found two more important parameters (i.e., \( lCd \) and \( mD \)) than the Kriging-based approach. This means the quasi-OTEE approach has a lower probability of yielding Type II errors, which is safer and preferred for the initial parameter screening.

Furthermore, if using \( \mu^* \) and \( S^T \) to sort the 3 most important parameters, it can be found that their ranks determined by the two approaches are also the same. As for the other parameters, some of them show different ranks depending on the approach used. For example, the parameter \( lCd \) is in the 4th position with the quasi-OTEE, but it is in the 8th position with the Kriging-based approach. However, if considering that the variances of the three most important parameters alone account for almost 80\% of the total variance of the results, the inconsistency of the ranks for the other parameters can be regarded as negligible, and the results from both approaches are still quite satisfactory.

To sum up, this experiment shows that the two approaches are able to identify the most important parameters as well as their ranks to a good degree, but with very low computational cost. Therefore, both approaches are recommended for the SA of computationally expensive traffic simulation models, especially when the other quantitative SA techniques (such as the standard variance-based SA) are unfeasible.

### 4.4 Summary of the Chapter

In this chapter a comparison of two recently developed SA approaches is made to better understand their advantages and disadvantages, especially in the context of computationally expensive microscopic traffic simulation models. The results show that both approaches are able to identify, to a good degree, the important and non-important parameters. The Kriging-based approach makes the model simpler. It is able to define with very high precision
the ranking of the most important parameters, but at the cost of missing some high dimensional interactions. On the other hand, the quasi-OTEE approach is very robust on all types of interactions, and therefore it is better in identifying those non-important parameters to discard.

These findings open the door to a possible alternative approach for the SA of computationally expensive traffic simulation models: the quasi-OTEE SA can be used for screening the parameters at the first stage, and decide which parameters to discard. Then, the Kriging-based SA can be used to refine the analysis and calculate the first order and the total sensitivity indexes, so that the correct rank of the important parameters can be obtained. In this way, it is expected that just a few hundreds simulations can produce results as accurate and reliable as any other more computationally expensive SA technique such as the variance-based SA. Therefore, it is worth exploring the development of a sequential SA approach that combines the quasi-OTEE and the Kriging-based approach. The next chapter is devoted to demonstrate the application of such sequential approach for complex models.
Chapter 5

Sequential Sensitivity Analysis for Model with Independent Parameters

This chapter is a combination of the findings from:


5.1 Introduction

The study in Chapter 4 shows that the Kriging-based SA is an efficient method for the quantitative SA of computationally expensive models. It is able to achieve approximately the same sensitivity indexes as those calculated adopting the standard variance-based approach. However, the main limitation of the Kriging-based SA is its dependence on the dimension of the model. When the model is high dimensional, the estimation of the Kriging metamodel using the Gaussian process may still be problematic due to the high computational cost.

On the other hand, case studies in Chapters 3 and 4 demonstrate that the quasi-OTEE SA is an efficient approach in identifying influential and non-influential parameters from a high dimension model. It yields fewer Type II errors in parameter screening, which is ideal for reducing the model dimension at a very early stage. Thus, to fully exploit the own strengths of Quasi-OTEE SA and Kriging-based SA, it is reasonable and practical to sequentially apply these two approaches: the quasi-OTEE SA is used first to screen the influential and non-influential parameters of a high dimensional model; then the Kriging-based SA is used to calculate the Sobol’ sensitivity indexes, and to rank the most influential parameters in a more accurate way.

The application of the proposed sequential SA is illustrated with two case studies. The first case study performs experiments based on several high dimensional numerical functions. Results show that the method can properly identify the most influential parameters and their ranks, while the number of model evaluations is considerably less than the standard variance-based SA (e.g., in one of the tests the sequential SA requires over 50 times fewer model evaluations than the standard variance-based SA). The second case study applies the sequential SA on the Wiedemann-74 car-following model developed by Wiedemann (1974), which is a commonly used car-following model for microscopic simulators such as Vissim (PTV, 2012). The proposed sequential SA method is applied to screen the most influential parameters out of 25 parameters of the Wiedemann-74 model. Results show that the sequential SA method performs similarly as the standard variance-based SA, but it requires 40 times fewer model evaluations.

The rest of this chapter is organized as follows. Section 5.2 briefly introduces the methodology of the sequential SA method. Section 5.3 presents the case studies, i.e., the SA of numerical functions and the SA of Wiedemann-74 car-following model. The experimental design for the sequential SA is introduced in detail, and the results from the two case studies are discussed respectively. Section 5.4 concludes this chapter with a summary of the corresponding findings.

5.2 Sequential Sensitivity Analysis

Chapter 4 compares the quasi-OTEE and the Kriging-based approach, in which these two SA approaches are both applied to two different microscopic traffic simulation models, i.e., Aimsun and Vissim. These two traffic models are considered as black-boxes, thus no specific assumptions regarding their internal mechanisms are made when performing the SA. Yet, they are rather complex simulation models in which correlations and interactions do exist among the inputs. The successful application of the two SA approaches on them is therefore
rather meaningful, and the corresponding results can be considered as general indicators on the performance of both SA methods.

In particular, the findings in Chapter 4 indicate that both the quasi-OTEE and the Kriging-based approach are able to identify well the influential parameters of the model. Compared to the Kriging-based approach, the quasi-OTEE has a higher accuracy in identifying the most influential parameters. Thus, the quasi-OTEE is better for the preliminary parameter screening, as it is less likely to discard the influential parameters in the first stage of SA. On the other hand, the Kriging-based approach has a higher precision in ranking the influential parameters. Hence, the Kriging-based approach is preferable for refining the screening results, and identifying the correct rank of the most influential parameters.

Therefore, it is meaningful to combine the quasi-OTEE and the Kriging-based approach as a sequential SA approach\(^1\). The simulation model is first analyzed through the quasi-OTEE SA, in which the influential and non-influential parameters are identified. Then the Kriging-based approach is used to derive the metamodel based only on the variations of the influential parameters (i.e., the non-influential parameters are fixed to nominal values when estimating the metamodel). At last, the variance-based SA is performed based on the Kriging metamodel to determine the variance contributions of the influential parameters as well as their ranks.

Moreover, recall that the total computational time required by the variance-based SA for deriving the sensitivity indexes is \(N(k + 2) \times t\), where \(t\) is the computation time for each run of the model (i.e., simulation model or metamodel). With the sequential SA, the initial parameter screening via quasi-OTEE reduces the number of relevant parameters considered in the final variance-based SA, i.e., \(k\) is significantly reduced from the beginning. Moreover, as the computational cost for running the metamodel can be regarded as negligible when the simulation model is computationally expensive, \(t\) is also greatly reduced after replacing the simulation model with the Kriging metamodel in the variance-based SA. If compared directly with the standard variance-based SA which directly performs the Monte Carlo experiments with the simulation model, it is obvious that the sequential SA can achieve higher computational benefits. More details about the performance of the sequential SA can be found in the following case studies, in which the proposed approach is applied to analyze different numerical test functions and the Wiedemann-74 car-following model.

## 5.3 Case Studies

### 5.3.1 Case study 1: numerical test functions

In this case study, numerical experiments are performed to demonstrate and test the application of the proposed sequential SA approach, and to validate its accuracy in comparison to other SA approaches such as the variance-based SA. To this end, four different test functions, which are commonly used as benchmarks in evaluating different SA approaches in e.g.,

---

\(^1\) The sequential SA approach presented in this chapter is dedicated for models that only contain independent parameters. In other words, only the main variances and the variances caused by parameter interactions are considered in this approach. For the analysis of the variance caused by parameter dependence, which is usually much more challenging, two general but more sophisticated SA approaches are provided in Chapters 7 and 8. Specifically, an extended sequential SA for models containing both independent and dependent parameters is presented in Section 8.3.2.
are used. Moreover, all these functions are, loosely speaking, high dimensional functions with more than 10 parameters.

5.3.1.1 Test functions

(1) G function

The G function (Archer et al., 1997) is a strongly non-linear and non-monotonic function (see the examples in Saltelli et al., 2010). It can be defined as:

\[ G = \prod_{i=1}^{k} g_i(X_i), \]  
\[ g_i(X_i) = \frac{|4X_i - 2| + a_i}{1 + a_i}. \]

The parameters \( X_i (i = 1, 2, \cdots, k) \) are uniformly distributed in the \( k \)-dimensional uniform interval \( U^k[0, 1] \). The coefficients \( a_i (i = 1, 2, \cdots, k) \) are non-negative real numbers, and they determine the variations of the corresponding functions \( g_i(X_i) \). In general, a high value of \( a_i \) indicates a low variation of \( g_i(X_i) \); on the other hand, a low value of \( a_i \) (e.g., \( a_i = 0 \)) means a high first order effect of \( X_i \).

In this experiment the sensitivities of the 12 parameters, i.e., \( X_1 \) to \( X_{12} \), of the G function are explored. Moreover, for the ease of demonstration and comparison, the same coefficients \( a_i (i = 1, 2, \cdots, 12) \) as those provided in Campolongo et al. (2007) are used (see Table 5.1). Although different values for these coefficients can be adopted in the test, it is expected that they will not substantially influence the main conclusion of this case study.

Table 5.1: Values of the 12 coefficients used in computing the G function.

<table>
<thead>
<tr>
<th>Coefficient</th>
<th>Value</th>
<th>Coefficient</th>
<th>Value</th>
<th>Coefficient</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>( a_1 )</td>
<td>0.001</td>
<td>( a_5 )</td>
<td>0.78</td>
<td>( a_9 )</td>
<td>74.51</td>
</tr>
<tr>
<td>( a_2 )</td>
<td>89.90</td>
<td>( a_6 )</td>
<td>1.26</td>
<td>( a_{10} )</td>
<td>4.32</td>
</tr>
<tr>
<td>( a_3 )</td>
<td>5.54</td>
<td>( a_7 )</td>
<td>0.04</td>
<td>( a_{11} )</td>
<td>82.51</td>
</tr>
<tr>
<td>( a_4 )</td>
<td>42.1</td>
<td>( a_8 )</td>
<td>0.79</td>
<td>( a_{12} )</td>
<td>41.62</td>
</tr>
</tbody>
</table>

(2) G* function

The G* function (Saltelli et al., 2010) is a modified version of the G function with additional shift and curvature terms. It is also a strongly non-linear and non-monotonic function. The G* function is defined as:

\[ G^* = \prod_{i=1}^{k} g_i^*(X_i), \]  
\[ g_i^*(X_i) = \frac{(1 + \alpha_i) \cdot |2 \cdot (X_i + \delta_i - I[X_i + \delta_i]) - 1|^{\alpha_i} + a_i}{1 + a_i}. \]
Same as the G function, all parameters $X_i (i = 1, 2, \cdots, k)$ of the $G^*$ function are assumed to have uniform distributions in the $k$-dimensional uniform interval $\mathcal{U}^k[0, 1]$. The coefficients $a_i \geq 0$ are also used to describe the variations of $g^*_i(X_i)$. Again, the higher the value of $a_i$, the smaller the variation of $g^*_i(X_i)$. $\alpha_i$ are the curvature parameters ($\alpha_i > 0$), and $\delta_i$ are the shift parameters ($0 \leq \delta_i \leq 1$). $I[X_i + \delta_i]$ is the integer part of $X_i + \delta_i$. It is obvious that when $\alpha_i = 1$ and $\delta_i = 0$, $g^*_i(X_i)$ is the same as $g_i(X_i)$.

In this test, the $G^*$ function with 20 parameters, i.e., $X_1$ to $X_{20}$, is employed. Moreover, two different sets of the coefficients $a_i$ and $\alpha_i$ are considered, so that the $G^*$ function can have different numbers of influential parameters. Again, for the ease of comparison, the same coefficients $a_i$ and $\alpha_i$ ($i \in [1, 20]$) as those presented in Campolongo et al. (2011) are used (see Table 5.2). The study by Saltelli et al. (2010) showed that the shift coefficients $\delta_i$ do not have any influence in computing the sensitivity indexes of the $G^*$ function, $\delta_i$ will be assigned values randomly picked from the uniform interval $[0, 1]$ for all tests.

Table 5.2: Values of the 20 coefficients used in computing the $G^*$ function.

<table>
<thead>
<tr>
<th>Index</th>
<th>Set-1</th>
<th></th>
<th>Set-2</th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>$a$</td>
<td>$\alpha$</td>
<td>$a$</td>
<td>$\alpha$</td>
</tr>
<tr>
<td>1</td>
<td>100</td>
<td>1</td>
<td>100</td>
<td>1</td>
</tr>
<tr>
<td>2</td>
<td>0</td>
<td>4</td>
<td>0</td>
<td>4</td>
</tr>
<tr>
<td>3</td>
<td>100</td>
<td>1</td>
<td>100</td>
<td>1</td>
</tr>
<tr>
<td>4</td>
<td>100</td>
<td>1</td>
<td>100</td>
<td>1</td>
</tr>
<tr>
<td>5</td>
<td>100</td>
<td>1</td>
<td>100</td>
<td>1</td>
</tr>
<tr>
<td>6</td>
<td>100</td>
<td>1</td>
<td>100</td>
<td>1</td>
</tr>
<tr>
<td>7</td>
<td>1</td>
<td>0.5</td>
<td>1</td>
<td>0.4</td>
</tr>
<tr>
<td>8</td>
<td>0</td>
<td>3</td>
<td>10</td>
<td>3</td>
</tr>
<tr>
<td>9</td>
<td>100</td>
<td>1</td>
<td>0</td>
<td>0.8</td>
</tr>
<tr>
<td>10</td>
<td>100</td>
<td>1</td>
<td>0</td>
<td>0.7</td>
</tr>
<tr>
<td>11</td>
<td>0</td>
<td>2</td>
<td>9</td>
<td>2</td>
</tr>
<tr>
<td>12</td>
<td>100</td>
<td>1</td>
<td>0</td>
<td>1.3</td>
</tr>
<tr>
<td>13</td>
<td>100</td>
<td>1</td>
<td>100</td>
<td>1</td>
</tr>
<tr>
<td>14</td>
<td>100</td>
<td>1</td>
<td>100</td>
<td>1</td>
</tr>
<tr>
<td>15</td>
<td>1</td>
<td>0.5</td>
<td>4</td>
<td>0.3</td>
</tr>
<tr>
<td>16</td>
<td>100</td>
<td>1</td>
<td>100</td>
<td>1</td>
</tr>
<tr>
<td>17</td>
<td>100</td>
<td>1</td>
<td>100</td>
<td>1</td>
</tr>
<tr>
<td>18</td>
<td>0</td>
<td>1.5</td>
<td>7</td>
<td>1.5</td>
</tr>
<tr>
<td>19</td>
<td>100</td>
<td>1</td>
<td>100</td>
<td>1</td>
</tr>
<tr>
<td>20</td>
<td>1</td>
<td>0.5</td>
<td>2</td>
<td>0.6</td>
</tr>
</tbody>
</table>

(3) K function

The K function is introduced in Bratley et al. (1992), and used in e.g., Campolongo et al. (2007), Saltelli et al. (2010), and Kucherenko et al. (2011) for testing and comparing different
SA approaches. It is defined as:

$$K = \sum_{i=1}^{k} (-1)^i \prod_{j=1}^{i} X_j,$$

(5.3)

where $X_i$ ($i = 1, 2, \cdots, k$) are model parameters that are all uniformly distributed in the $k$-dimensional uniform interval $U^k[0, 1]$.

According to Saltelli et al. (2010), the parameters of the K function do not have the same influence on the output (i.e., the variation of some parameters dominates the variation of the output), and there are significant interaction effects among the parameters with higher indexes (Campolongo et al., 2007).

This test employs the K function with 20 parameters, i.e., $X_1$ to $X_{20}$. According to Campolongo et al. (2007), $X_1$ and $X_2$ are the two parameters that have the highest total effect.

(4) Morris function

The Morris function (Morris, 1991) is defined as:

$$M = \beta_0 + \sum_{i=1}^{20} \beta_i w_i + \sum_{i<j}^{20} \beta_{i,j} w_i w_j$$

$$+ \sum_{i<j<l}^{20} \beta_{i,j,l} w_i w_j w_l + \sum_{i<j<l<s}^{20} \beta_{i,j,l,s} w_i w_j w_l w_s,$$

(5.4a)

$$w_i = \begin{cases} \frac{2.2X_i}{X_i + 0.1} - 1, & \text{if } i = 3, 5, 7 \\ 2(X_i - 0.5), & \text{otherwise} \end{cases}$$

(5.4b)

This function has 20 parameters, i.e., $X_1$ to $X_{20}$. These parameters are uniformly distributed in the interval $U^{20}[0, 1]$. In Equation (5.4a), $\beta_i$, $\beta_{i,j}$, $\beta_{i,j,l}$, and $\beta_{i,j,l,s}$ are respectively the first-, second-, third- and fourth-order coefficients. The coefficients with relatively high values in the Morris function are:

$$\beta_i = 20 \text{ when } i = 1, 2, \cdots, 10,$$

(5.5a)

$$\beta_{i,j} = -15 \text{ when } i, j = 1, 2, \cdots, 6,$$

(5.5b)

$$\beta_{i,j,l} = -10 \text{ when } i, j, l = 1, 2, \cdots, 5,$$

(5.5c)

$$\beta_{i,j,l,s} = 5 \text{ when } i, j, l, s = 1, 2, \cdots, 40.$$

(5.5d)

All other first- and second-order coefficients are independently picked from the multivariate standard normal distribution, i.e., $\mathcal{N}(0, 1)$. The remaining third- and fourth-order coefficients are set to 0. According to the findings in Morris (1991), the parameters in the Morris function can be grouped into three categories:

1) influential parameters with high interaction effects;
2) influential parameters with low interaction effects; and
3) non-influential parameters.
In the test the proposed sequential SA is used to check the sensitivity of each parameter in order to find the corresponding category, and calculate the first order and total sensitivity indexes.

It should be noted that the test functions described above are not strictly computationally expensive models. The reason for using these functions in the numerical experiment is because the true sensitivity indexes of these functions can be obtained via either analytical calculation or applying other SA techniques such as a standard variance-based SA. In this way, it is possible to directly compare the results from the sequential SA with the true SA results, and hence prove the accuracy of the sequential SA. In any case, whether the model is computationally expensive or not, the process of the proposed sequential SA should remain the same, although the benefits from the proposed SA are more clearly visible with computationally expensive models.

5.3.1.2 Experimental design

In the sequential SA, the quasi-OTEE approach is first applied to screen the influential parameters from the non-influential parameters. Since the test functions in this study all have low computational costs, it is feasible to use a high number of quasi-OT for sampling. Therefore, for each test function 100 quasi-OTs are selected from 1,000 randomly generated Morris trajectories. The computational cost of the quasi-OTEE in the numerical experiment is derived as $100(k + 1)$ where $k$ is the number of model parameters.

In the next step, based on the screening results from the quasi-OTEE analysis, the Kriging-based approach is used to rank the influential parameters. For each test function, the influential parameters are sampled using the Sobol’ quasi random sequence (Sobol’, 1976), while the non-influential parameters found by the quasi-OTEE are fixed to 0.5 (i.e., the mean of the uniform distribution $U[0, 1]$). Considering that the computational cost of the test function itself is low, in this study 1,024 evaluations are performed for each test function to derive the corresponding Kriging metamodel. Note that the number of evaluations, i.e., 1,024, is further shown to be sufficient to provide a reliable metamodel based on the approach mentioned in Ciuffo et al. (2013).

For the quasi-OTEE SA, the influential parameters are selected based on the sensitivity indexes $\mu$ and $\sigma$ (see the selection rule in Section 3.2.1). For the Kriging-based approach, final results are presented with the first order sensitivity index and the total sensitivity index (i.e., $S$ and $S^T$). To assess and compare the accuracy of the sequential SA, the sensitivity indexes $S$ and $S^T$ for the G, $G^*$, and K functions are analytically calculated using the formulas in Table 5.3 (Saltelli et al., 2010). As there is no analytical formula for computing the sensitivity indexes $S$ and $S^T$ for the Morris function, $S$ and $S^T$ are estimated numerically with the variance-based SA (see Equations (2.19) and (2.20)).

Table 5.4 reports the computational cost (i.e., the number of simulation model runs) required by the sequential SA and the variance-based SA for each test function in this study. It is worth to recall that the number of simulation model runs in estimating the Kriging metamodel is considered as the computational cost of the Kriging-based approach. For the variance-based SA, it starts with a small number of model evaluations, and this number is gradually increased in order to check the stability of the SA results. Table 5.4 only reports the
Table 5.3: Analytical formulations for the first order and total sensitivity indexes for the G, $G^*$ and K functions. Recall that $S_i = V_i / V$ and $S^T_i = V^T_i / V$.

<table>
<thead>
<tr>
<th>Function</th>
<th>Variance</th>
<th>Analytical Formula</th>
</tr>
</thead>
<tbody>
<tr>
<td>$V_i$</td>
<td>$V^T_i$</td>
<td>G</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>G</td>
<td>$1$</td>
<td>$3(1 + a_i)^2$</td>
</tr>
<tr>
<td></td>
<td>$1$</td>
<td>$3(1 + a_i)^2 + 1</td>
</tr>
<tr>
<td></td>
<td></td>
<td>$j=1$</td>
</tr>
<tr>
<td></td>
<td></td>
<td>$1 + \frac{1}{3(1 + a_j)^2}$</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>$k$</td>
<td></td>
</tr>
<tr>
<td></td>
<td>$k$</td>
<td>$1 + \frac{1}{3(1 + a_j)^2}$ - 1</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>$V_i$</td>
<td>$a_i^2$</td>
<td>$G^*$</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>$V^T_i$</td>
<td>$a_i^2$</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>$(1 + 2a_i)(1 + a_i)^2$</td>
</tr>
<tr>
<td></td>
<td></td>
<td>$j=1$</td>
</tr>
<tr>
<td></td>
<td></td>
<td>$1 + \frac{a_j^2}{(1 + 2a_j)(1 + a_j)^2}$</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>$k$</td>
<td></td>
</tr>
<tr>
<td></td>
<td>$k$</td>
<td>$1 + \frac{a_j^2}{(1 + 2a_j)(1 + a_j)^2}$ - 1</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>$V_i$</td>
<td>$\frac{1}{27} \left[ 2 \left( -\frac{1}{2} \right)^i + \left( -\frac{1}{2} \right)^k \right]^2$</td>
<td>K</td>
</tr>
<tr>
<td></td>
<td>$\frac{1}{8} \left( \frac{1}{3} \right)^i + \frac{1}{40} \left( \frac{1}{3} \right)^k + 10 \left( \frac{2}{3} \right)^i \left( -\frac{1}{2} \right)^k$</td>
<td></td>
</tr>
<tr>
<td></td>
<td>$\frac{1}{10} \left( \frac{1}{3} \right)^k - \frac{1}{9} \left( \frac{1}{2} \right)^2k - 2 \left( -\frac{1}{2} \right)^k + \frac{1}{18}$</td>
<td></td>
</tr>
</tbody>
</table>

Note: $i$ is the index of the parameter, and $k$ is the number of parameters contained in the corresponding function.

minimum computational cost required by the variance-based SA to achieve stable sensitivity indexes in this case study.

Table 5.4: Number of original model runs for deriving the sensitivity indexes of the test functions.

<table>
<thead>
<tr>
<th>Function</th>
<th>Number of parameters</th>
<th>Sequential SA</th>
<th>Variance-based SA</th>
</tr>
</thead>
<tbody>
<tr>
<td>G</td>
<td>12</td>
<td>1,300 (quasi-OTEE) + 1,024 (Kriging)</td>
<td>–</td>
</tr>
<tr>
<td>$G^*$</td>
<td>20</td>
<td>2,100 (quasi-OTEE) + 1,024 (Kriging)</td>
<td>–</td>
</tr>
<tr>
<td>K</td>
<td>20</td>
<td>2,100 (quasi-OTEE) + 1,024 (Kriging)</td>
<td>–</td>
</tr>
<tr>
<td>Morris</td>
<td>20</td>
<td>2,100 (quasi-OTEE) + 1,024 (Kriging)</td>
<td>180,224</td>
</tr>
</tbody>
</table>

Finally, it is necessary to underline that the objective of the above analysis is to understand whether the proposed approach can provide sufficient accuracy without analyzing the minimum number of model runs required. It is possible that the computational cost presented here is higher than the minimum computational cost. However, the analysis of the minimum number of model runs would be very much dependent on the features of the test functions (in
particular their regularity within the input space), and therefore it is considered beyond the scope of this case study which employs generic test functions.

5.3.1.3 Results of case study 1

The results of the quasi-OTEE SA for all tests are shown in Figures 5.1(a), 5.2(a), 5.3(a), 5.4(a), and 5.5(a). The Kriging-based SA is performed based on the screening results of the quasi-OTEE SA, and the results are shown in Figures 5.1(b), 5.2(b), 5.3(b), 5.4(b), and 5.5(b). Moreover, the SA results obtained via analytical calculation for the G, G* and K function, and via the standard variance-based SA for the Morris function, are also included in these figures as references for demonstrating the performance of the Kriging-based SA. The details are given below.

(1) G function

It is clear that in Figure 5.1(a) seven parameters, \(X_1, X_3, X_5, X_6, X_7, X_8,\) and \(X_{10}\), have much higher values of \(\mu^*\) and \(\sigma\) than the other five parameters. Therefore, parameters \(X_1, X_3, X_5, X_6, X_7, X_8,\) and \(X_{10}\) are selected as the influential parameters, and they are further analyzed by the Kriging-based SA. In addition, parameters \(X_1\) and \(X_7\) have the highest \(\mu^*\) and \(\sigma\), which means they have high interaction effects with other parameters. On the other hand, the \(\mu^*\) and \(\sigma\) of parameters \(X_2, X_4, X_9, X_{11},\) and \(X_{12}\) are very close to 0. This indicates that the variations of these parameters have very low impacts on the variation of the model output, and hence they can be fixed without influencing the results of the subsequent Kriging-based SA.

Figure 5.1(b) shows the total sensitivity indexes \(S^T\) and the first-order sensitivity indexes \(S\) derived by both the Kriging-based SA and the analytical calculation for the G function. According to the Kriging-based SA, it is clear that parameters \(X_1\) and \(X_7\) are the two most influential parameters. Each of them accounts alone for about 30%, and in combination with other parameters for about 45% of the variations of the model output. The sensitivity indexes derived by the Kriging-based SA for \(X_1\) and \(X_7\) are consistent with the indexes obtained through the analytical calculation. This confirms that the Kriging-based SA can reach a high accuracy in analyzing the sensitivity of the most influential parameters.

The third to fifth most influential parameters determined by the Kriging-based SA are \(X_8, X_5\) and \(X_6\). Note that according to the analytical calculation, the \(S^T\) of \(X_5\) is slightly larger than the \(S^T\) of \(X_8\), which implies that \(X_5\) is actually more influential than \(X_8\). However, since the share of variances caused by these two parameters is fairly low, and the differences in the \(S^T\) between \(X_8\) and \(X_5\) are trivial, the error in ranking these two parameters by the Kriging-based SA is expected to have insignificant impacts on the final results. Such inaccuracy is, in any case, not a surprise. As already presented in Chapter 4, the Kriging-based approach is very accurate in defining the sensitivity of the most influential parameters, although it may present some deficiencies for parameters with a minor effect on the outputs.

The other two parameters, \(X_3\) and \(X_{10}\), have the lowest \(S^T\) among the 7 parameters (i.e., less than 3%), which means the variations of these two parameters have very limited impacts on the variations of the model output, and hence they can also be regarded as non-influential parameters.
Figure 5.1: SA results of the G function.

(a) Results of the quasi-OTEE SA for the G function.

(b) Sensitivity indexes of the influential parameters obtained by the Kriging-based SA and analytical calculation for the G function.

(2) G* function

The quasi-OTEE results of the G* function with respect to the two coefficient sets (see Table 5.2) are shown in Figures 5.2(a) and 5.3(a).

Figure 5.2(a) shows that there are 7 influential parameters of the G* function with Coefficient Set-1, i.e., X_2, X_7, X_8, X_11, X_15, X_18, and X_20. Moreover, parameters X_2, X_8, and X_18 are the most influential parameters and they have high interaction effects with other parameters. It is worth noting that compared with the screening results for the G* function that uses the same coefficient set in Campolongo et al. (2011), all the 4 influential parameters in Campolongo et al. (2011), i.e., X_2, X_8, X_11, and X_18, are included in the screening results of the quasi-OTEE approach, while the quasi-OTEE SA has found 3 additional influential parameters. This confirms the accuracy of the quasi-OTEE approach in parameter screening, and its capability in reducing the Type II error. On the other hand, it also shows that this approach alone is not sufficient if the objective is to get the minimum possible number of factors, and it is not effective in avoiding Type I error, although the Type I error may cause considerably less problems than the Type II error.

Figure 5.3(a) presents the quasi-OTEE results of the G* function under Coefficient Set-2. It is obvious that 10 parameters, i.e., X_2, X_7, X_8, X_9, X_10, X_11, X_12, X_15, X_18, and X_20, have high values of $\mu^*$ and $\sigma$. Therefore, these 10 parameters are regarded as the influential
Figure 5.2: SA results of the $G^*$ function with Coefficient Set-1.

(a) Results of the quasi-OTEE SA for the $G^*$ function with Coefficient Set-1.

(b) Sensitivity indexes of the influential parameters obtained by the Kriging-based SA and analytical calculation for the $G^*$ function with Coefficient Set-1.

parameters in this case. Moreover, these 10 parameters are exactly the same influential parameters found in Campolongo et al. (2011) for the $G^*$ function with the same coefficient set. The other 10 parameters have significantly low values of $\mu^*$ and $\sigma$, which implies that they are non-influential parameters, and they will be fixed during the subsequent Kriging-based SA.

The results of the Kriging-based SA for the $G^*$ function using the two different coefficient sets are presented respectively in Figures 5.2(b) and 5.3(b). In Figure 5.2(b), it is obvious that parameters $X_2$, $X_8$, $X_{11}$ and $X_{18}$ have much higher $S^T$ than the other 3 parameters. Hence, parameters $X_2$, $X_8$, $X_{11}$, and $X_{18}$ are the most influential parameters for the $G^*$ function with Coefficient Set-1. It is worth mentioning that for the 4 most influential parameters, although the $S^T$ derived by the Kriging-based SA are all slightly lower than the corresponding analytic $S^T$, the Kriging-based SA gives exactly the same parameters ranking as the analytical calculation. This again proves the good performance of the Kriging-based SA in determining the ranking of the most influential parameters. The other 3 parameters $X_7$, $X_{15}$, and $X_{20}$ can be regarded as non-influential parameters since their total impact on the variation of the model output is less than 10%.

For the $G^*$ function with Coefficient Set-2, Figure 5.3(b) indicates that the most influential parameter in this case is $X_2$, which accounts alone for almost 35%, and in combination with other parameters for almost 80% of the variations of the model output. The second to fourth most influential parameters determined by the Kriging-based SA are $X_{12}$, $X_{10}$, and $X_9$, respectively.
Figure 5.3: SA results of the $G^*$ function with Coefficient Set-2.

(a) Results of the quasi-OTEE SA for the $G^*$ function with Coefficient Set-2.

(b) Sensitivity indexes of the influential parameters obtained by the Kriging-based SA and analytical calculation for the $G^*$ function with Coefficient Set-2.

respectively. Although the analytical calculation gives opposite rankings for $X_{10}$ and $X_9$, as the absolute differences in $S_T^*$ for these two parameters are not significant (i.e., less than 2%), the SA results from the Kriging-based approach are still valid and satisfactory. Furthermore, since the variations of the other 6 parameters, i.e., $X_7$, $X_8$, $X_{11}$, $X_{15}$, $X_{18}$ and $X_{20}$, only cause very limited variations of the model output (less than 10% in total), these 6 parameters are accordingly identified as non-influential parameters in this case.

(3) K function

The screening results for the K function are presented in Figure 5.4(a). The quasi-OTEE SA has found 7 influential parameters, i.e., $X_1$, $X_2$, $X_3$, $X_4$, $X_5$, $X_6$, and $X_7$, because they have significantly higher values of $\mu^*$ and $\sigma$ than the other 13 parameters. Note that in Campolongo et al. (2011) only parameters $X_1$, $X_2$, and $X_3$ are chosen as the influential parameters. This again highlights the ability of the quasi-OTEE SA to avoid the Type II error. As explained before, this feature is preferable for the parameter screening at the beginning.

Figure 5.4(b) illustrates the Kriging-based SA results for the K function. Parameter $X_1$ appears to be the most influential parameter of the K function. It accounts alone for over 65%, and in combination with other parameters for almost 75% of the variations of the model output. The second and third most influential parameters are $X_2$ and $X_3$, which account for
Figure 5.4: SA results of the K function.

(a) Results of the quasi-OTEE SA for the K function.

(b) Sensitivity indexes of the influential parameters obtained by the Kriging-based SA and analytical calculation for the K function.

almost 25% and 10% of the total variations of the model output, respectively. The share of the variances caused by the other 4 parameters, i.e., $X_4, X_5, X_6,$ and $X_7,$ is less than 5% in total, therefore they are regarded as non-influential parameters. It is worth underlining that in this case, the Kriging-based SA yields almost the same $S^T$ and $S$ as the analytical calculation for all 7 parameters, which again highlights the accuracy of the Kriging-based SA.

(4) Morris function

Figure 5.5(a) illustrates the quasi-OTEE SA results for the Morris function. Here 10 parameters are chosen as the influential parameters, i.e., $X_1, X_2, \ldots, X_{10},$ due to their high values in terms of $\mu^*$ and $\sigma.$ Among these 10 parameters, parameters $X_8, X_9,$ and $X_{10}$ will have strong linear but low interaction effects with other parameters since they all have high $\mu^*$ but low $\sigma.$ The other 7 influential parameters, i.e., $X_1, X_2, \ldots, X_7,$ will have non-linear and strong interaction effects with other parameters because of the high values of $\sigma.$ The other 10 parameters of the Morris function, i.e., $X_{11}, X_{2}, \ldots, X_{20},$ are regarded as non-influential parameters according to the quasi-OTEE analysis.

Figure 5.5(b) illustrates the $S$ and $S^T$ for the 10 parameters of the Morris function estimated by the Kriging-based SA and the standard variance-based SA. The most influential parameters are $X_1, X_2,$ and $X_4,$ and each of them accounts for almost 25% of the total variations in the
Chapter 5. Sequential Sensitivity Analysis for Model with Independent Parameters

Figure 5.5: SA results of the Morris function.

(a) Results of the quasi-OTEE SA for the Morris function.

(b) Sensitivity indexes of the influential parameters obtained by the Kriging-based SA and variance-based SA for the K function.

The above numerical experiments have shown that the proposed sequential SA has satisfactory accuracy. In the test of the G, G* and K functions, the sequential SA has derived almost the same S and ST for the influential parameters as the analytical calculations. This has confirmed that the sequential SA has a very high accuracy for both qualitative SA (e.g., identifying the influential parameters and their ranks) and quantitative SA (e.g., deriving the sensitivity indexes) of a high dimensional model. Such feature is especially useful for the SA of the high-dimensional model output. The 4th to 10th most influential parameters of the Morris function are X9, X10, X8, X3, X5, X6, and X7. In addition, since the first 6 parameters (i.e., X1, X2, ..., X6) have low S (almost 0) but high ST, these parameters are expected to have high interaction effects. On the other hand, parameters X7, X8, X9, and X10 have almost the same S and ST, therefore they can be identified as influential parameters with low interaction effects.

Furthermore, when comparing the S and ST derived by the Kriging-based SA and the standard variance-based SA for each parameter individually, it can be seen that the results are very close to each other. Hence, in this case, the Kriging-based SA has the same performance as the standard variance-based SA in terms of the accuracy of the results, while having much lower computational cost (see the comparison of the computational cost in Table 5.4).

5.3.1.4 Summary of the SA results from case study 1

The above numerical experiments have shown that the proposed sequential SA has satisfactory accuracy. In the test of the G, G* and K functions, the sequential SA has derived almost the same S and ST for the influential parameters as the analytical calculations. This has confirmed that the sequential SA has a very high accuracy for both qualitative SA (e.g., identifying the influential parameters and their ranks) and quantitative SA (e.g., deriving the sensitivity indexes) of a high dimensional model. Such feature is especially useful for the SA of the high-dimensional model output. The 4th to 10th most influential parameters of the Morris function are X9, X10, X8, X3, X5, X6, and X7. In addition, since the first 6 parameters (i.e., X1, X2, ..., X6) have low S (almost 0) but high ST, these parameters are expected to have high interaction effects. On the other hand, parameters X7, X8, X9, and X10 have almost the same S and ST, therefore they can be identified as influential parameters with low interaction effects.

Furthermore, when comparing the S and ST derived by the Kriging-based SA and the standard variance-based SA for each parameter individually, it can be seen that the results are very close to each other. Hence, in this case, the Kriging-based SA has the same performance as the standard variance-based SA in terms of the accuracy of the results, while having much lower computational cost (see the comparison of the computational cost in Table 5.4).
dimensional and computationally expensive models or commercial models, which usually behave like black-boxes and do not allow for analytical derivation of the model sensitivities (e.g., first-order derivative).

Moreover, it is worth highlighting that the results of the sequential SA even though achieved with only 3,124 model evaluations (i.e., 2,100 evaluations for quasi-OTEE, and 1,024 evaluations for estimating the Kriging metamodel), are comparable with those achieved with 180,224 model evaluations using the standard variance-based SA. In other words, the proposed approach required over 50 times fewer model evaluations. Although the Morris model used here is not a computationally expensive model, when considering the huge difference in the number of model runs required by the sequential SA and the variance-based SA, it is obvious that the sequential SA for a computationally expensive model can bring significant benefits in terms of a reduced computational cost.

5.3.2 Case study 2: Wiedemann-74 car-following model

In this case study the proposed sequential SA approach is applied to the Wiedemann-74 car-following model. The aim is to illustrate the working process and outcome of the proposed sequential SA, especially in the context of a traffic simulation model.

5.3.2.1 Wiedemann-74 car-following model

Wiedemann-74 model (Wiedemann, 1974; Wiedemann and Reiter, 1992) is a well-known car-following model used in microscopic traffic simulations (e.g., Vissim). It belongs to the psychophysical car-following models (Michaels, 1963), i.e., the driving behavior is determined by specific driving regimes for the leading vehicle (i.e., leader) and the following vehicle (i.e., follower). Four driving regimes are considered in this model, i.e., free driving, closing, following, and emergency braking (see Figure 5.6).

The model has 7 inputs which represent the vehicle kinematics in the car-following process, i.e., positions ($x_l$ and $x_f$, the subscripts $l$ and $f$ indicate the leader and follower respectively), speeds ($v_l$ and $v_f$), acceleration rates ($acc_l$ and $acc_f$), and the length of the leading vehicle ($L_l$).

In Figure 5.6, one of the four driving regimes is assigned to the follower according to:
i) its relative speed (i.e., $\Delta v$, $\Delta v = v_f - v_l$) and relative position (i.e., $\Delta x$, $\Delta x = x_l - x_f - L_l$) with respect to the leader in the current time step, which are described by the 7 kinematic inputs; and
ii) the thresholds for $\Delta v$ and $\Delta x$ which are defined by the model parameters. The physical meaning of the corresponding thresholds in Figure 5.6 are:

$ABX$ is the desired minimum following distance at low speed differences.

$AX$ is the desired safety distance for standing vehicles (front-to-front distance).

$BX$ is the additional minimum following distance when driving.

$SDX$ is the maximum following distance.
Figure 5.6: Schematic drawing of the Wiedemann-74 model with different driving regimes and thresholds.


**SDV** is the perception threshold of speed difference at long distances.

**CLDV** is the perception threshold of speed difference at short, decreasing distances. CLDV is similar to SDV, but it has a much wider range of variation (Wiedemann, 1974).

**OPDV** is the perception threshold of speed difference at short, but increasing distances.

Depending on the driving regime, the follower’s reaction to the stimuli coming from the leader will vary. In this case study, the predicted acceleration rate of the follower in the next time step is used as the model output. Figure 5.7 shows the calculation process of the acceleration rate of the follower (i.e., \( acc_f(t + 1) \)) based on the inputs in the current time step (i.e., \( t \)) in the four driving regimes.

---

\(^{2}\) The perception threshold of speed difference represents the point at which the follower consciously realizes that he is driving faster/slower than the leader in a car-following process.
Figure 5.7: Calculation process of the acceleration in the Wiedemann-74 car-following model.

The corresponding formulations in Figure 5.7 are described below:

\[ AX = L_t + AX_1 + AX_2 \times R_1 \]  
\[ BX = (BX_1 + BX_2 \cdot R_1) \cdot \sqrt{\min\{v_f(t), v_l(t)\}} \]  
\[ SDV(t) = \left[ \frac{\Delta x(t) - AX}{CX_0 \times (CX_1 + CX_2 \times (R_1 + R_2))} \right]^2 \]  
\[ SDX(t) = AX + BX \times EX \]  
\[ EX = EX_1 + EX_2 \times (NR - R_2) \]  
\[ CLDV(t) = SDV(t) \times EX^2 \]  
\[ OPDV(t) = CLDV(t) \times (-OPDV_1 - OPDV_2 \times NR) \]  
\[ b_{null} = BNULL_1 \times (R_3 + NR) \]  
\[ b_{max}(t) = BMAX_1 \times [VMAX - v_f(t) \times F] \]  
\[ F = \frac{VMAX}{VDES + F_1 \times (VMAX - VDES)} \]  
\[ b_{min}(t) = -BMIN_1 - BMIN_2 \times [R_4 - v_f(t)] \]  

where \( b_{null} \) is the boundary value of acceleration and deceleration, \( b_{max}(t) \) is a vehicle’s maximum acceleration rate, and \( b_{min} \) is a vehicle’s maximum deceleration rate. \( AX_1, AX_2, R_1, BX_1, BX_2, RX_0, CX_1, CX_2, R_2, EX_1, EX_2, NR, OPDV_1, OPDV_2, BNULL_1, R_3, VMAX \) (i.e., maximum speed of the follower), \( VDES \) (i.e., desired speed of the follower), \( BMAX_1, F_1, BMIN_1, BMIN_2, R_4 \) and \( K \) are the 24 model parameters (Wiedemann, 1974) which are adjustable in model calibration.

In addition, the parameters \( R_1, R_2, R_3 \) and \( R_4 \) are four driver dependent random parameters for describing the stochastic driving behaviors (e.g., driving ability, safety needs, and aggressiveness) across different drivers. They are all normally distributed in the interval \([0, 1]\) with mean 0.5 and standard deviation 0.15. The parameter \( NR \) is a random number to model the variation of \( SDX \) for the same driver (Wiedemann and Reiter, 1992). It is also normally distributed in the interval \([0, 1]\) with mean 0.5 and standard deviation 0.15. The parameter \( K \) is a model parameter to model the learning process in the closing regime. For example, when the conscious deceleration lasts longer, the follower has a better estimate of the distance to the leader. The value of \( K \) is incremented at every simulation second.

It is worth mentioning that unlike some other car-following models (e.g., the Gipps model) which uses an explicit parameter to define the driver’s reaction time, the reaction time is modeled based on other parameters and inputs in the Wiedemann-74 model. In this model, the follower tries to maintain his kinematic status within a certain driving regime (although some stochastic variations of the speed and acceleration are also considered), and only take actions (e.g., changes the acceleration rate) when \( \Delta v \) and \( \Delta x \) cross some certain thresholds, i.e., when the driving regime is changed.

For example, suppose that there is a scenario where the follower and leader are in the following regime. When the leader starts accelerating, the follower is still assigned to the following regime. He will try to maintain his current kinematic status and hence no action is taken. After some simulation steps, when \( \Delta v \) and \( \Delta x \) are big enough to reach the thresholds (i.e., \( OPDV \) and \( SDX \)), he will be assigned to the free driving regime, and will try to reach the
desired speed within this regime. If his desired speed is higher than his current speed in the following regime, he will start to accelerate. Therefore, the time spent in those simulation steps in order to make the follower change his driving regime is just the reaction time, although not explicitly defined in this model. In addition, since the thresholds mentioned above are defined by the 24 model parameters, variations of model parameters will bring variations to the thresholds, and then to the reaction time. As a result, different drivers will have different reaction times for different driving regimes. Thus, a study which analyzes the impacts from the variations of different model parameters, also explores the impacts due to the variations of reaction time, although this is done in an indirect way.

5.3.2.2 Experimental design

In this case study, only the acceleration rate is considered as the model output. Although for real simulations there are other model outputs (e.g., traffic flow, travel time), as the proposed approach is a general SA approach (i.e., it is independent of any specific model or data type), the different model outputs will not necessarily change the SA steps described here.

The review in the previous section shows that Wiedemann-74 model is a high dimensional model, and a general calibration for all 24 model parameters will be rather difficult and time consuming. Thus, the main objective of SA is to reduce the model dimension and optimize the calibration. Through quantifying the contributions of the 24 parameters on output variations, non-influential parameters are eliminated from the calibration, and priorities can be set for the influential parameters. Thus, the 24 model parameters are all included in the SA. In particular, in the model the maximum speed of a vehicle (i.e., $V_{MAX}$) should always be greater than its desired speed (i.e., $V_{DES}$). However, during the SA the values for $V_{MAX}$ and $V_{DES}$ are independently sampled, which may result in $V_{MAX}$ being less than $V_{DES}$. To avoid this problem, the parameter $V_{MAX}$ is replaced by a new parameter $V_{DIFF}$ in the parameter sampling. In the model calculation $V_{MAX}$ is derived as the sum of $V_{DIFF}$ and $V_{DES}$.

In a single simulation no variability of the parameters is considered. In different simulations, the values assigned to all parameters are extracted from uniform distributions whose upper and lower bounds are shown in Table 5.5. Because the 5 random parameters $R_1$, $R_2$, $R_3$, $R_4$ and $NR$ are normally distributed with $\bar{N}(0.5, 0.15)$ in the interval [0, 1], in this case study they are uniformly and independently sampled in the interval [0.05, 0.95] to stay within the 99% of possible values of the original normal distribution.

Furthermore, since the variability of driver and vehicle also bring significant variations to the model output, it may happen that some parameters are influential with a certain combination of kinematic inputs, but they might be non-influential under the other kinematic settings. Therefore, the SA results will be meaningless if they are just obtained under a fixed kinematic setting. For this reason, the 7 kinematic inputs (i.e., $x_l$, $x_f$, $v_l$, $v_f$, $acc_l$, $acc_f$ and $L_l$) should also be included as SA factors, i.e., their variations are considered in the SA. Moreover, when some kinematic inputs are identified as influential inputs by the SA, although they are not adjustable in model calibration, it is expected that the uncertainties of model output can be reduced if the uncertainties of these kinematic inputs are reduced in advance. Therefore, to increase the simulation quality, it is reasonable to perform the SA to investigate the impacts of different kinematic inputs together with the model parameters.

Because in the car-following process the 7 kinematic inputs are correlated with each other,
Chapter 5. Sequential Sensitivity Analysis for Model with Independent Parameters

Table 5.5: 25 factors for the sensitivity analysis of the Wiedemann-74 model.

<table>
<thead>
<tr>
<th>#</th>
<th>Name</th>
<th>Range</th>
<th>#</th>
<th>Name</th>
<th>Range</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>AX₁</td>
<td>[0.01, 2]</td>
<td>14</td>
<td>BMIN₁</td>
<td>[0.1, 8]</td>
</tr>
<tr>
<td>2</td>
<td>AX₂</td>
<td>[0.01, 2]</td>
<td>15</td>
<td>BMIN₂</td>
<td>[0.01, 0.1]</td>
</tr>
<tr>
<td>3</td>
<td>BX₁</td>
<td>[0.5, 4]</td>
<td>16</td>
<td>BNULL₁</td>
<td>[0.01, 10]</td>
</tr>
<tr>
<td>4</td>
<td>BX₂</td>
<td>[0.01, 5]</td>
<td>17</td>
<td>VDIFF</td>
<td>[5, 10]</td>
</tr>
<tr>
<td>5</td>
<td>CX₀</td>
<td>[0.1, 50]</td>
<td>18</td>
<td>VDES</td>
<td>[10, 20]</td>
</tr>
<tr>
<td>6</td>
<td>CX₁</td>
<td>[0.5, 10]</td>
<td>19</td>
<td>R₁</td>
<td>[0.05, 0.95]</td>
</tr>
<tr>
<td>7</td>
<td>CX₂</td>
<td>[0.01, 10]</td>
<td>20</td>
<td>R₂</td>
<td>[0.05, 0.95]</td>
</tr>
<tr>
<td>8</td>
<td>EX₁</td>
<td>[0.5, 10]</td>
<td>21</td>
<td>R₃</td>
<td>[0.05, 0.95]</td>
</tr>
<tr>
<td>9</td>
<td>EX₂</td>
<td>[0.01, 10]</td>
<td>22</td>
<td>R₄</td>
<td>[0.05, 0.95]</td>
</tr>
<tr>
<td>10</td>
<td>OPDV₁</td>
<td>[0.01, 10]</td>
<td>23</td>
<td>NR</td>
<td>[0.05, 0.95]</td>
</tr>
<tr>
<td>11</td>
<td>OPDV₂</td>
<td>[0.01, 10]</td>
<td>24</td>
<td>K</td>
<td>[1, 100]</td>
</tr>
<tr>
<td>12</td>
<td>F₁</td>
<td>[0.01, 10]</td>
<td>25</td>
<td>Kin</td>
<td>[1, 1024]</td>
</tr>
<tr>
<td>13</td>
<td>BMAX₁</td>
<td>[0.1, 0.2]</td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

and the sequential SA proposed in this chapter works only with models that have independent inputs, the kinematic inputs are grouped as an independent input Kin in this case study. Each value assigned to the kinematic input Kin in the interval presented in Table 5.5 defines a possible combination of the 7 kinematic inputs. As the methodology for computing the sensitivity indexes is the same for both model parameters and the input Kin, for the ease of explanation, they are termed as factors hereafter in this case study. Overall, 25 factors are considered in the SA, including 24 model parameters and 1 kinematic group input Kin.

It should be noted that for demonstration purposes, the ranges shown in Table 5.5 are arbitrarily chosen to be wider than their normal values, hence a complete SA can be performed in the whole input space (i.e., the SA can explore all possible values of the model inputs, including those values in very rare situations). In practice, the data ranges might be narrowed by referencing e.g., field measurement data, and recommendations by other empirical studies. However, the SA procedure proposed in this chapter should remain the same.

In order to have a representative set of combinations among the 7 kinematic inputs, real data collected from six car-following experiments in different road/traffic conditions are employed in this case study. These experiments were carried out along roads in the area of Naples, Italy, under real traffic conditions between October 2002 and July 2003. Experiments were performed by driving four vehicles along urban and inter-urban roads under different traffic conditions (always without lane-changing). All vehicles were equipped with Kinematic GPS receivers, which recorded the position of each vehicle at 0.1 second intervals. Post data processing included the differential correction of raw GPS coordinates, with the aid of data gathered by a fifth stationary receiver acting as “base station”, and through the application of an elaborate filtering procedure. More details about the experimental set-up and the data gathering can be found in (Punzo et al., 2005; Punzo and Simonelli, 2005).

---

3 Chapter 7 provides an improved variance-based SA which is able to solve the SA problem with dependent parameters. In that chapter the sensitivities of the 7 kinematic inputs can be analyzed separately.
Moreover, the car-following data are available for public access from the MULTITUDE project \cite{MULTITUDE}. From these data, a total of 16,425 car-following observations (i.e., each observation is a combination of the 7 kinematic inputs) are included in the present study. The sensitivity analysis is performed 50 times to investigate the effect of the combinations used for these seven variables, and to increase the generality and the robustness of the results. Each sensitivity analysis considered a subset with 1,024 observations randomly extracted from the original data set. At the end of the analysis, the distributions of the sensitivity measures are studied so it could be determined whether the results are sufficiently robust and general.

The 50 SA are carried out using both the standard variance-based SA and the sequential SA proposed in this study. In this way, it is possible to understand the advantages of the sequential SA approach. The 50 standard variance-based SA are based on 442,368 model evaluations each (for a total of more than 22 million evaluations), while the 50 sequential SA only required 181,200 simulations in total (3,624 each). The huge number of simulations required by the variance-based SA is the main reason for choosing this “computationally cheap” car-following model rather than a more complex simulation package in this case study.

In the sequential SA, the quasi-OTEE approach is first applied to screen the influential and non-influential factors. In particular, 100 quasi-OT are selected from 1,000 randomly generated Morris trajectories (requiring 2,600 model evaluations). Then in the next step, based on the screening results of the quasi-OTEE, the Kriging-based approach is employed to find the correct rank of the most influential factors (requiring 1,024 model evaluations). Finally, the distributions of the results across all 50 experiments are investigated. For the quasi-OTEE approach, the selection of influential factors is based on the \( \mu \), \( \mu^* \), and \( \sigma \) of the corresponding EEs. For the Kriging-based approach, final results are presented in the form of median, 5th and 95th percentile of the distributions of both \( S \) and \( S^T \). Such criteria are also used for the standard variance-based SA.

### 5.3.2.3 Results of case study 2

Results are presented first for the standard variance-based SA and then for the sequential SA.

#### (1) Standard variance-based sensitivity analysis

Results of the standard variance-based SA are presented in Figure 5.8. It is clear that the most influential factor is the combination of kinematic inputs sampled for each evaluation (i.e., \( Kin \)). It accounts alone for 50\% and, in combination with the other factors, for 85\% of the variability of the acceleration value derived from the model. The second most influential factor is \( BMI N_1 \), which drives the deceleration phase and accounts for 20\% of the variance in combination with the other factors. The 3rd to 10th most influential factors are \( EX_1, N_R, R_2, BX_1, R_1, CX_0, BX_2, \) and \( EX_2 \). The effect of the rest of the factors can be considered negligible, as they mainly have influences only in combination with the other factors.

Results therefore show that the variability of the model output is influenced 50\% by the kinematic inputs of the model (which are described by the factor \( Kin \)), and 50\% by the model parameters (although this effect is still largely in combination with \( Kin \)). Limiting the model calibration to the 8 most influential factors would lead to a well-balanced model, whose
Figure 5.8: Results of the standard variance-based SA for the Wiedemann-74 model. Median, 5th, and 95th percentile of the distributions of the first order and the total sensitivity indices are derived from the 50 variance-based SA of the full set of 25 factors of the Wiedemann-74 model.
application is still influenced by the kinematic conditions and by a reasonable number of factors.

Figure 5.8 also shows that the variation of the indexes among different experiments is relatively low (i.e., the variation does not change the rank of the different factors). In addition, an analysis is carried out to check the numerical stability of the corresponding sensitivity indexes via increasing the number of model evaluations (see Punzo et al., 2015, for more details). It is found that the results are sufficiently stable after 135,000 out of the 442,368 model evaluations. This indicates that 135,000 simulations are the minimum number of simulations required by the standard variance-based approach in this case.

(2) Sequential sensitivity analysis

Results based on the quasi-OTEE analysis are shown in Figure 5.9. Figure 5.9(a) shows the plots of the $\mu^*$ and $\sigma$ of the EE for the 25 factors. The K-means clustering analysis (MacQueen et al., 1967) is applied to group the factors with similar $\mu^*$ and $\sigma$ of the EE. Three clusters can be identified accordingly. It is clear that Kin has the highest $\mu^*$ and $\sigma$, hence it is the most influential factor. The factors in Cluster 3 have the lowest $\mu^*$ and $\sigma$, therefore they are the least influential factors.

To examine the rest of the factors, a wedge formed by two lines with respect to $\mu = \pm 2\sigma/\sqrt{n}$ ($n$ is the sample size, see Chapter 3, $n = 100$ in this case) is employed in the $\mu$-$\sigma$ plot in Figure 5.9(b). According to Morris (1991), the variation of any factor lying outside the wedge will surely influence the model output, independently of its adopted value, the range of the variation, and the adopted values of the other factors. Therefore, BMIN1, EX1, VDES, BX1, NR, CX0, BNULL1, R2, and BMAX1 (although BMAX1 belongs to Cluster 3 in Figure 5.9(a)) are also considered as influential factors at this stage. A comparison between Figures 5.9(a) and 5.9(b) shows BX2, R1, and EX2 have low $\mu$ but relatively high $\mu^*$ and $\sigma$, hence they will have both positive and negative effects depending on the values of other factors. To reduce the Type II error, they are also identified as influential factors at this stage, although this will require extra computation efforts in the subsequent Kriging-based analysis.

Overall, 13 factors, i.e., Kin, BMIN1, EX1, VDES, BX1, NR, CX0, BNULL1, BX2, R2, R1, EX2 and BMAX1, are selected as the influential factors from the quasi-OTEE. Notice that all the 8 influential inputs from the standard variance-based SA are included in the screening results of the quasi-OTEE SA. This again confirms the good performance of this method, and its capability to avoid Type II error (i.e., in this case the Type II error is completely eliminated).

Results of the Kriging-based approach for the 13 important factors mentioned above are presented in Figure 5.10. They show that the variability of the indexes is much higher than that in the standard variance-based SA. This is certainly due to the fact that with less model evaluations, the results are more sensitive to the different sets of kinematic inputs combinations considered. In addition, results also show that the distribution of $S^T$ is not symmetric, with a higher dispersion towards the higher values. It is therefore necessary to take this into consideration for the factor selection.

The Kriging-based approach also finds that the most influential factor is Kin, which accounts for 50% and 85% of the variability in the acceleration value, alone by itself and in combination with the other factors, respectively. The second most influential factor is
Figure 5.9: Screening results of the sequential SA for the Wiedemann-74 car-following model.

(a) Plots of $\mu^*$ versus $\sigma$ of the EE for the 25 factors. The plots are separated into 3 clusters based on the K-Means Clustering method.

(b) Plots of $\mu$ versus $\sigma$ of the EE for the 25 factors. Lines in the figure correspond to $\mu = \pm 2SEM$.

$BM1N_1$, which accounts for 20% of the output variance. According to Figure 5.10 and the analysis of the distributions of the $S$ and $S^T$, the following factors are identified as influential factors as well: $NR$, $BX_1$, $EX_1$, $BMAX_1$, $BNUL1$, $R_1$, $BX_2$ and $EX_2$, for a total of 10 factors. This final selection would result in two Type I errors (i.e., $BNUL1$ and $BMAX_1$) and two Type II errors (i.e., $R_2$ and $CX_0$). However, the share of variance caused by these factors is fairly low and therefore, these errors are expected to have no significant effects on the outputs of the model. Such inaccuracy is not unexpected. As already discussed in Chapter 4, the Kriging-based approach is very accurate in defining the sensitivity of the most influential factors, although it may present some deficiencies for factors with a small effect on the outputs.

Furthermore, the above case study also shows that the Kriging-based approach is able to provide quantitative information regarding the relations among factors. For example, according to Equation (5.6b), $BX_1$ could cause twice the output variance as $BX_2$ because the
mean value of $R_1$ is 0.5. This outcome is consistent with the SA results from Figure 5.10: the black bar (i.e., the total sensitivity index) for $BX_1$ is almost twice as long as that for $BX_2$. Such feature is especially useful for the quantitative SA of many commercial simulation models (e.g., Vissim, Aimsun), which usually behave like black-boxes and do not allow for calculation of the first-order derivatives.

It is worth highlighting that the results of the sequential SA, although achieved with only 3,624 model evaluations, are fairly comparable with those achieved with 135,000 model evaluations using the standard variance-based SA. In other words, the sequential SA requires approximately 40 times less model evaluations in this case study. Compared with the standard variance-based approach, the sequential SA for a traffic simulation package can reach a good balance between feasibility and accuracy.

Finally, it should be noted that the above case study is intended only to illustrate the application of the sequential SA, so the results may not necessarily be transferred to other simulation studies, even when the same car-following model is used. On the other hand, as the sequential SA approach is developed as a general SA tool (i.e., it is independent of any specific traffic simulation model or data type), it is fully transferable to other traffic simulation studies. In other words, no matter how the model is built or which data type (e.g., aggregated or disaggregated) is used in the simulation, the sequential SA and its methodology described
in this chapter should always remain the same.

5.4 Summary of the Chapter

This chapter provides an efficient and practical SA method, which sequentially combines the quasi-OTEE and the Kriging-based approach for the SA of high dimensional and/or computationally expensive models. The influential parameters are first screened by the quasi-OTEE approach. Then based on the screening results, the Kriging-based SA is applied to further identify the rank of the most influential parameters.

If compared with a stand-alone screening method that can only provide qualitative information about influential and non-influential parameters, the sequential SA is able to produce more quantitative results such as the first order and the total sensitivity indexes. In addition, thanks to the parameter screening by the quasi-OTEE, the number of parameters considered in estimating the Kriging metamodel is greatly reduced. As a result, it is more efficient than a stand-alone Kriging-based method discussed in Chapter 4 when dealing with a high dimensional model.

The performance of the sequential SA is empirically demonstrated via two case studies. The first case study performs four numerical experiments based on widely used test functions. The results in all tests have shown that, the proposed method can properly identify the most influential parameters and their ranks. The results are consistent with those obtained through analytical calculations and a standard variance-based SA, while the number of model evaluations required by the sequential SA is considerably less than that required by the variance-based SA (e.g., in the test of Morris function, the sequential SA requires over 50 times less model evaluations than the variance-based SA). The second case study is based on the Wiedemann-74 car-following model. The results also show that the sequential SA is able to correctly derive the ranks of the most influential factors from the 25 factors of the model, while the number of model evaluations is 40 times lower than that needed by the standard variance-based SA.

This feature makes the sequential SA very attractive, especially when dealing with high dimensional and computationally expensive models that could require several hours or even a day per run (as a matter of fact, according to Saltelli (2002), the variance-based SA is affordable only for models whose cost per run is less than some minutes, see Table 2.1). Furthermore, since the proposed method is a general and global SA method independent of any specific model, there should be no difficulty to transfer this method as a standard SA tool for any model in the wider scientific community. In other words, no matter if the model is complex or not, the sequential SA approach and methodology described in this chapter will remain the same.

One specific requirement for applying the above sequential SA is the model should contain independent parameters only. To solve the SA problem for models that contain both dependent and independent parameters, the SA techniques discussed here need to be extended accordingly. The details are introduced in the following chapters.
Chapter 6

Sampling Approach for Dependent Parameters

This chapter contains the findings from the following research paper:

6.1 Introduction

Most SA methods and guidelines for traffic simulation models reviewed in Chapter 2 are designed to work with models containing independent parameters only. As a result, many practitioners have to perform the SA by either assuming that the model parameters are independent beforehand (Punzo and Ciuffo, 2014b), or simply grouping the dependent parameters as one independent parameter (see the case study in Section 5.3.2).

However, in practice, due to certain constrains in the input space, and/or the intricate relations among the parameters obtained from empirical experiments (e.g., some parameters are the outcome of another model or experiment, see Brell et al., 2010), it is very likely that many parameters are actually dependent parameters, or mixtures of both independent and dependent parameters. Thus, simply assuming all parameters are independent and using the SA methods dedicated to independent parameters could yield incorrect results. For instance, some parameters could be identified as non-influential parameters if they are treated as independent in the SA. However, if they have strong correlations with those influential parameters, these “non-influential” parameters may actually be influential parameters.

Therefore, given the fact that there are usually many dependent parameters contained in a traffic simulation model (e.g., the speed and the acceleration rate of a vehicle), the research of an efficient and accurate SA approach for dependent parameters is very important. In Kucherenko et al. (2012) and Marà et al. (2015), the classic Sobol’ formulas (Sobol’, 1993) are extended, so that they can be used to derive the sensitivity indexes for the dependent parameters (see Chapter 7 for more details). However, computing these sensitivity indexes analytically can be very challenging or even unfeasible when the model is a complex model or a black-box (this is especially true for most commercial simulators such as Vissim and Aimsun). As an alternative, the sensitivity indexes can be estimated in a Monte Carlo framework based on a large number of dependent random samples. In other words, the dependent random samples are required for performing the SA of dependent parameters.

To this end, this chapter presents a general approach for generating random samples of dependent parameters. This approach does not only consider the marginal distributions of individual parameters, but also takes the dependence structure of all model parameters into account. Specifically, the Gaussian copula and the Spearman’s rank correlation coefficient (Iman and Conover, 1982; Mara and Tarantola, 2012) are employed in the sampling process. Moreover, a case study is included to demonstrate the proposed sampling approach, which generates dependent random samples for the kinematic inputs of the Wiedemann-74 car-following model presented in Section 5.3.2.

The rest of this chapter is organized as follows: the methodology of the sampling approach for dependent parameters is described in Section 6.2; the application of the proposed sampling algorithm for dependent parameters is illustrated with a case study in Section 6.3; the conclusions and suggestions for future work are included in Section 6.4.

6.2 Methodology

In order to generate random samples for dependent parameters, it is required that the corresponding parameters have known marginal distributions and dependence structure. In
practice, the marginal distributions are usually obtained from literature, relevant research, empirical data, expert opinions, etc. (Punzo and Ciuffo, 2014b). In case no a priori information is available, the marginal distribution function of a parameter can be chosen based on reasonable hypotheses. For instance, it is often assumed that the corresponding parameter could have a uniform distribution in a reasonable range (e.g., ±50% of the default value). The dependence structure (e.g., the correlation matrix) can be obtained in the same way as the case of the marginal distributions. If no information is available, the practitioner may choose to perform the sampling and the SA by assuming the parameters are independent, and later validate the correctness of such hypothesis (Punzo and Ciuffo, 2014b). More details about defining the marginal distributions and dependent structure are provided in Section 9.2.2.

In many cases the parameters of a traffic simulation model can have different types of marginal distributions that are not necessarily standard distributions (e.g., uniform, normal, lognormal). In order to generate dependent samples with arbitrary marginal distributions and a known dependence structure, the Gaussian copula (Nelsen, 1999) can be adopted. The details are introduced below.

### 6.2.1 Gaussian copula

Let $F_i(\cdot)$ be the marginal Cumulative Distribution Function (CDF) of parameter $X_i$, and $F_i^{-1}(\cdot)$ be the inverse CDF of $X_i$. For a model $f$ with $k$ independent parameters, i.e., $\{X_1, \cdots, X_k\}$, the joint CDF of these $k$ independent parameters is the product function of the marginal CDFs of all parameters:

$$F_{1,\cdots,k}(x_1, \cdots, x_k) = \mathbb{P}\left(\{X_1 \leq x_1\} \cap \cdots \cap \{X_k \leq x_k\}\right) = \mathbb{P}(X_1 \leq x_1) \times \cdots \times \mathbb{P}(X_k \leq x_k) = \prod_{i=1}^{k} F_i(x_i), \quad (6.1)$$

where $\mathbb{P}(\cdot)$ stands for the probability.

In the case of dependent parameters, the joint CDF of $\{X_1, \cdots, X_k\}$ can also be written as a function (not a simple product function because of the correlation of the parameters) of the marginal CDFs:

$$F_{1,\cdots,k}(x_1, \cdots, x_k) = C\left(F_1(x_1), \cdots, F_k(x_k); \Sigma_x\right) = C(u^c_1, \cdots, u^c_k; \Sigma_u), \quad (6.2)$$

where $\Sigma_x$ is the covariance matrix of $\{X_1, \cdots, X_k\}$. $\{u^c_1, \cdots, u^c_k\}$ are correlated random variables with uniform marginal distributions in $U^k[0, 1]$ and covariance matrix $\Sigma_u$.

The function $C(\cdot)$ is known as a copula (for details see Nelsen, 1999). It is defined as the joint CDF of random variables $\{U_1, \cdots, U_k\}$ with uniform marginal distributions $U[0, 1]$:

$$C(u^c_1, u^c_2, \cdots, u^c_k; \Sigma_u) = \mathbb{P}\left(\{U_1 \leq u^c_1\} \cap \cdots \cap \{U_k \leq u^c_k\}\right). \quad (6.3)$$

Copula has recently become quite popular in fields such as risk management, quantitative finance, and civil engineering (Caniou, 2012). It can be used to generate multivariate distributions for modelling the dependence structure of correlated multivariate data. There
are many different types of copulas, and one commonly used copula for sampling dependent parameters is the Gaussian copula\(^1\):

\[
C^\text{Gauss}_k(u^c_1, \ldots, u^c_k; \Sigma_u) = \Phi_k \left( \Phi^{-1}(u^c_1), \ldots, \Phi^{-1}(u^c_k); \Sigma_z \right) = \Phi_k(z^c_1, \ldots, z^c_k; \Sigma_z),
\]

(6.4)

where \(\Phi^{-1}(\cdot)\) is the inverse CDF of the univariate standard normal distribution, and \(\Phi_k(\cdot)\) is the joint CDF of the \(k\)-dimensional standard normal distribution. \(\{z^c_1, \ldots, z^c_k\}\) are correlated random variables with standard normal marginal distributions in \(N^k(0, \Sigma_z)\) and covariance matrix \(\Sigma_z\).

Combining Equations (6.2) and (6.4), the following equality can be derived:

\[
F_{1,\ldots,k}(x_1, \ldots, x_k) = C^\text{Gauss}_k(F_1(x_1), \ldots, F_k(x_k); \Sigma_x) = \Phi_k \left( \Phi^{-1}(F_1(x_1)), \ldots, \Phi^{-1}(F_k(x_k)); \Sigma_z \right) = \Phi_k(z^c_1, \ldots, z^c_k; \Sigma_z).
\]

(6.5)

Accordingly, \(z^c_i = \Phi^{-1}(u^c_i) = \Phi^{-1}(F_i(x_i))\) for \(\forall i \in [1, k]\). Then by applying the inverse transformation, the following equation can be obtained:

\[
x_i = F_i^{-1} \left( \Phi(z^c_i) \right) \quad \forall i \in [1, k].
\]

(6.6)

The above transformation is also known as the inverse Nataf transformation (Liu and Der Kiureghian, 1986; Der Kiureghian and Liu, 1986; Li et al., 2008). It indicates that the model parameters \(\{X_1, \ldots, X_k\}\) with any arbitrary marginal CDFs \(\{F_1(\cdot), \ldots, F_k(\cdot)\}\) can be obtained by first sampling the correlated base vector \(\{z^c_1, z^c_2, \ldots, z^c_k\}\) from the \(k\)-dimensional standard normal distribution \(N^k(0, \Sigma_z)\), and then applying the component-wise transformation \(T_i(\cdot) = F_i^{-1}(\Phi(\cdot))\) to this base vector. It is worth mentioning that the reason for choosing multivariate normal random vectors as the base vectors in the transformation is because they can be easily generated (Ghosh and Henderson, 2002). In practice, it is also possible to employ random vectors generated from other distributions if different copulas are used to model the dependence structure.

Moreover, the above copula-based sampling does not require any knowledge about the conditional or joint distribution functions of model parameters. These functions are normally much harder (sometimes even unfeasible) to be obtained in practice if compared with the marginal distribution functions. Therefore, the copula-based sampling is more general and practical than the Markov Chain Monte Carlo sampling (Saltelli, 2002; Kanso et al., 2006), in which the conditional or joint distribution functions are required.

---

\(^1\)Note that a multivariate distribution is not uniquely determined by the marginal distributions or correlations. In practice, other copulas may be chosen in order to fit the specific dependence structure (e.g., see Zou and Zhang, 2016; for a recent work that employs multiple copulas to model the dependence structure of microscopic traffic variables). However, the sampling process described here is the same.
6.2.2 Spearman’s rank correlation

The transformation \( F_i^{-1}(\Phi_i(\cdot)) \) is monotonic, but in most cases it is non-linear. Thus, if the dependence structure is described by the linear correlation, \( \Sigma_z \) needs to be carefully selected so that the random sample could have the desired covariance matrix (i.e., \( \Sigma_x \)). This can be done using empirical estimators (e.g., Liu and Der Kiureghian, 1986; Der Kiureghian and Liu, 1986) or algorithms (e.g., Li et al., 2008). However, when the model parameters have arbitrary marginal distributions and/or non-linear correlations, the selection of a proper \( \Sigma_z \) can become rather challenging or sometimes even unfeasible.

To solve the above problem, as an alternative, the rank correlation can be used to describe the dependence structure of the model parameters. Specifically, Spearman’s Rank Correlation Coefficient (SRCC, see Kendall, 1970) is used in this dissertation. Note that in practice it is also possible to employ other rank correlation coefficients such as Kendall’s Tau (Kendall, 1970). The formula for computing SRCC is given below:

\[
\rho_{X_i, X_j}^{s} = 1 - \frac{6}{n(n^2-1)} \sum_{o=1}^{n} (R^o_{X_i} - R^o_{X_j})^2 ,
\]

where \( R^o_{X_i} \) stands for the rank (in the ascending order) of the \( o \)-th observation of parameter \( X_i \), and \( n \) is the total number of observations. For example, if \( X_i \) has three observations \{4.1, 0.1, 2.8\}, then \( \{R^1_{X_i}, R^2_{X_i}, R^3_{X_i}\} = \{3, 1, 2\} \).

The rank is an invariant measure under monotonic transformations, thus the rank correlation coefficient is also invariant through any monotonic transformation, regardless of the linearity of the transformation. In the above Gaussian copula, because both \( F_i^{-1}(\cdot) \) and \( \Phi(\cdot) \) are monotonic functions, the model parameters \( \{X_i, X_j\} \) and the corresponding random normal samples \( \{z^c_i, z^c_j\} \) should have the same rank correlation coefficient, i.e., \( \rho_{X_i, X_j}^{r} = \rho_{z^c_i, z^c_j}^{r} \) for \( \forall \{i, j\} \in [1, k] \).

In addition, according to Hotelling and Pabst (1936), the following equality holds for multivariate normal distribution:

\[
\rho_{z^c_i, z^c_j}^{r} = 2 \sin \left( \frac{\pi}{6} \rho_{X_i, X_j}^{s} \right),
\]

where \( \rho_{z^c_i, z^c_j}^{r} \) is the Linear Correlation Coefficient (LCC) for \( z^c_i \) and \( z^c_j \). Therefore, when the rank correlation between any two model parameters \( X_i \) and \( X_j \) is known, the elements in \( \Sigma_z \) can be computed as \( \rho_{z^c_i, z^c_j}^{r} = 2 \sin \left( \frac{\pi}{6} \rho_{X_i, X_j}^{s} \right) \).

6.2.3 Steps for generating dependent random samples

The process for generating random samples of dependent parameters \( \{X_1, \cdots, X_k\} \) with given arbitrary marginal CDFs \( \{F_1(\cdot), \cdots, F_k(\cdot)\} \) and rank correlations is summarized in the following steps:\footnote{This process is also known as Iman Conover (IC) procedure (Iman and Conover, 1982).}

1) Generate \( N \) independent random samples for each of the \( k \) parameters from the uniform distribution \( U(0,1) \). The random samples are presented in the matrix \( \tilde{U} \). The uniformly distributed random samples corresponding to parameter \( X_i \), i.e., the \( i \)-th column vector
\( \bar{U}_i \) in \( \bar{U} \), can be produced using pseudo-random number generators or low discrepancy sequences (e.g., Sobol’ sequence, see Sobol’, 1976). As a result, \( \bar{U}_1, \ldots, \bar{U}_k \) will be independent of each other.

\[
\bar{U} = \begin{bmatrix}
  u_{1,1} & u_{2,1} & \cdots & u_{k,1} \\
  u_{1,2} & u_{2,2} & \cdots & u_{k,2} \\
  \vdots & \vdots & \ddots & \vdots \\
  u_{1,N} & u_{2,N} & \cdots & u_{k,N}
\end{bmatrix},
\]

(6.9)

2) Apply the inverse CDF of the standard normal distribution, i.e., \( \Phi^{-1}(\cdot) \), to every element in \( \bar{U} \). The resulting matrix is \( \bar{Z} \):

\[
\bar{Z} = \begin{bmatrix}
  z_{1,1} & z_{2,1} & \cdots & z_{k,1} \\
  z_{1,2} & z_{2,2} & \cdots & z_{k,2} \\
  \vdots & \vdots & \ddots & \vdots \\
  z_{1,N} & z_{2,N} & \cdots & z_{k,N}
\end{bmatrix},
\]

(6.10)

where \( z_{i,r} = \Phi^{-1}(u_{i,r}) \), \( i \in [1, k] \), \( r \in [1, N] \). Obviously, the elements in any column vector \( \bar{Z}_i \) follow the standard normal distribution, and they are independent of the other column vectors.

3) Compute the SRCC \( \rho_{X_i, X_j}^s \) for all pairs of \( \{X_i, X_j\} \) using Equation (6.7). Due to the symmetry of the covariance matrix, it is only required to compute the SRCC for the cases when \( 1 \leq i < j \leq k \). The covariance matrix \( \Sigma_z \) can be computed using Equation (6.8):

\[
\Sigma_z = \begin{bmatrix}
  1 & 2 \sin \left( \frac{\pi}{6} \rho_{X_1, X_2}^s \right) & \cdots & 2 \sin \left( \frac{\pi}{6} \rho_{X_1, X_k}^s \right) \\
  2 \sin \left( \frac{\pi}{6} \rho_{X_1, X_2}^s \right) & 1 & \cdots & 2 \sin \left( \frac{\pi}{6} \rho_{X_1, X_k}^s \right) \\
  \vdots & \vdots & \ddots & \vdots \\
  2 \sin \left( \frac{\pi}{6} \rho_{X_1, X_2}^s \right) & 2 \sin \left( \frac{\pi}{6} \rho_{X_2, X_k}^s \right) & \cdots & 1
\end{bmatrix}.
\]

(6.11)

4) Since \( \Sigma_z \) is a symmetric positive definite matrix, it can be decomposed as the product of a lower triangular matrix \( L \) and an upper triangular matrix \( L^T \) using the Cholesky decomposition:

\[
\Sigma_z = L \cdot L^T = \begin{bmatrix}
  l_{1,1} & 0 & \cdots & 0 \\
  l_{1,2} & l_{2,2} & \cdots & 0 \\
  \vdots & \vdots & \ddots & \vdots \\
  l_{1,k} & l_{2,k} & \cdots & l_{k,k}
\end{bmatrix} \begin{bmatrix}
  l_{1,1} & l_{1,2} & \cdots & l_{1,k} \\
  0 & l_{2,2} & \cdots & l_{2,k} \\
  \vdots & \vdots & \ddots & \vdots \\
  0 & 0 & \cdots & l_{k,k}
\end{bmatrix}.
\]

(6.12)
The findings are not surprising as in a car-following process, the follower will always adjust its speed and acceleration based on the speed and acceleration of the leader. Weak positive correlations between vehicle (i.e., Kin) and its speed and acceleration based on the speed and acceleration of the leader. These inputs are correlated with each other in the car-following process, while the aforementioned sequential approach can only deal with the SA of independent parameters. It has been found that Kin is the most influential factor, which accounts for over 85% of the total variance of the predicted acceleration rate. However, because the 7 kinematic inputs are jointly sampled as a group, it is not possible to identify which inputs are more important than the others. To investigate the impacts of individual kinematic inputs, the extended variance-based SA or the extended EE method can be adopted (see Chapters 7 and 8). These two methods are both based on QMC estimation, i.e., a large number of random samples for each individual parameter are required for deriving the corresponding sensitivity information. This case study illustrates how to apply the proposed sampling approach in order to generate random samples for the dependent kinematic inputs.

The marginal distributions, scatter plots, and SRCC of any two kinematic inputs are shown in Figure 6.1. The corresponding LCCs and SRCCs from the empirical data are reported in Table 6.1. Table 6.1 and Figure 6.1 show that a strong positive correlation exists between v_l and v_f. The pair acc_l-acc_f also shows certain amount of positive correlation. The findings are not surprising as in a car-following process, the follower will always adjust its speed and acceleration based on the speed and acceleration of the leader. Weak positive correlation between the original model inputs x_l, x_f and L_l in the sampling. This is done to avoid invalid samples in cases where x_l - L_l < x_f, which may happen if these three inputs are sampled separately. In total, five kinematic inputs, i.e., ∆x, v_f, v_l, acc_l, and acc_f, are considered in this case study.

The marginal distributions, scatter plots, and SRCC of any two kinematic inputs are shown in Figure 6.1. The corresponding LCCs and SRCCs from the empirical data are reported in Table 6.1. Table 6.1 and Figure 6.1 show that a strong positive correlation exists between v_l and v_f. The pair acc_l-acc_f also shows certain amount of positive correlation. The findings are not surprising as in a car-following process, the follower will always adjust its speed and acceleration based on the speed and acceleration of the leader. Weak positive correlation between the original model inputs x_l, x_f and L_l in the sampling. This is done to avoid invalid samples in cases where x_l - L_l < x_f, which may happen if these three inputs are sampled separately. In total, five kinematic inputs, i.e., ∆x, v_f, v_l, acc_l, and acc_f, are considered in this case study.

5) Generate the dependent random sample:

\[
\tilde{Z}^c = \tilde{Z} \cdot L^T = \begin{bmatrix}
z_1^c & z_2^c & \cdots & z_k^c \\
z_{1,1}^c & z_{2,1}^c & \cdots & z_{k,1}^c \\
z_{1,2}^c & z_{2,2}^c & \cdots & z_{k,2}^c \\
\vdots & \vdots & \ddots & \vdots \\
z_{1,N}^c & z_{2,N}^c & \cdots & z_{k,N}^c \\
\end{bmatrix}.
\]

(6.13)

6) Finally, the desired sample matrix \( \tilde{X} \) is obtained through component-wise transformation (see Equation (6.6)) of the corresponding element in \( \tilde{Z}^c \):

\[
\tilde{X} = \begin{bmatrix}
F_1^{-1}(\Phi(z_{1,1}^c)) & F_2^{-1}(\Phi(z_{2,1}^c)) & \cdots & F_k^{-1}(\Phi(z_{k,1}^c)) \\
F_1^{-1}(\Phi(z_{1,2}^c)) & F_2^{-1}(\Phi(z_{2,2}^c)) & \cdots & F_k^{-1}(\Phi(z_{k,2}^c)) \\
\vdots & \vdots & \ddots & \vdots \\
F_1^{-1}(\Phi(z_{1,N}^c)) & F_2^{-1}(\Phi(z_{2,N}^c)) & \cdots & F_k^{-1}(\Phi(z_{k,N}^c)) \\
\end{bmatrix}.
\]

(6.14)

6.3 Case Study

In the sequential SA study of the Wiedemann-74 model from Section 5.3.2, seven kinematic inputs, i.e., \( x_f, x_l, v_f, v_l, acc_f, acc_l \) and \( L_l \), are grouped as one single factor Kin in the analysis. This is because these inputs are correlated with each other in the car-following process, while the aforementioned sequential approach can only deal with the SA of independent parameters. It has been found that Kin is the most influential factor, which accounts for over 85% of the total variance of the predicted acceleration rate. However, because the 7 kinematic inputs are jointly sampled as a group, it is not possible to identify which inputs are more important than the others. To investigate the impacts of individual kinematic inputs, the extended variance-based SA or the extended EE method can be adopted (see Chapters 7 and 8). These two methods are both based on QMC estimation, i.e., a large number of random samples for each individual parameter are required for deriving the corresponding sensitivity information. This case study illustrates how to apply the proposed sampling approach in order to generate random samples for the dependent kinematic inputs.
Figure 6.1: Marginal distributions, scatter plots and SRCC of the five kinematic inputs based on empirical car-following data. Plots in the diagonal: marginal distribution of the corresponding input. Plots below the diagonal: scatter plots of the two inputs located in the corresponding row and column. Plots above the diagonal: SRCC of the two inputs located in the corresponding row and column.

Source: Punzo et al. (2005); Punzo and Simonelli (2005)
correlations can be found for inputs pairs $\Delta x - v_f$, $\Delta x - v_l$, as well as $acc_f - acc_l$. The correlation can be neglected for the rest of the kinematic input pairs. The empirical car-following data in Figure 6.1 also shows that it is very hard to use a standard multivariate distribution from statistics to describe the joint distribution of these kinematic inputs. This again highlights the need for the flexibility offered by copula in sampling data with arbitrary distributions.

Table 6.1: Correlation coefficients from the empirical data.

<table>
<thead>
<tr>
<th>Input Pairs</th>
<th>LCC</th>
<th>SRCC</th>
</tr>
</thead>
<tbody>
<tr>
<td>$\Delta x$ &amp; $v_f$</td>
<td>0.2716</td>
<td>0.1920</td>
</tr>
<tr>
<td>$\Delta x$ &amp; $v_l$</td>
<td>0.2717</td>
<td>0.1932</td>
</tr>
<tr>
<td>$\Delta x$ &amp; $acc_f$</td>
<td>0.0654</td>
<td>0.0711</td>
</tr>
<tr>
<td>$\Delta x$ &amp; $acc_l$</td>
<td>-0.0916</td>
<td>-0.0529</td>
</tr>
<tr>
<td>$v_f$ &amp; $v_l$</td>
<td>0.9847</td>
<td>0.9803</td>
</tr>
<tr>
<td>$v_f$ &amp; $acc_f$</td>
<td>0.0113</td>
<td>0.0290</td>
</tr>
<tr>
<td>$v_f$ &amp; $acc_l$</td>
<td>-0.1120</td>
<td>-0.0861</td>
</tr>
<tr>
<td>$v_l$ &amp; $acc_f$</td>
<td>0.1037</td>
<td>0.1056</td>
</tr>
<tr>
<td>$v_l$ &amp; $acc_l$</td>
<td>0.0014</td>
<td>0.0146</td>
</tr>
<tr>
<td>$acc_f$ &amp; $acc_l$</td>
<td>0.4769</td>
<td>0.4071</td>
</tr>
</tbody>
</table>

To generate the desired dependent random samples using the approach presented in Section 6.2, the first step is to generate independent random samples for each kinematic input with uniform distribution in $\mathcal{U}(0, 1)$, i.e., $\tilde{U} = [\tilde{U}_{\Delta x}, \tilde{U}_{v_f}, \tilde{U}_{v_l}, \tilde{U}_{acc_f}, \tilde{U}_{acc_l}]$. Then the inverse CDF of the standard normal distribution (i.e., $\Phi^{-1}()$) is applied to $\tilde{U}$ to generate independent random samples with standard normal distribution, i.e., $\tilde{Z} = \Phi^{-1}(\tilde{U}) = [\tilde{Z}_{\Delta x}, \tilde{Z}_{v_f}, \tilde{Z}_{v_l}, \tilde{Z}_{acc_f}, \tilde{Z}_{acc_l}]$.

In the next step, the SRCC of the empirical data are transformed into the LCC of the standard normal random samples (i.e., $\tilde{Z}^c$) by using Equation (6.11). The covariance matrix $\Sigma_z$ is shown below:

$$
\Sigma_z = \begin{bmatrix}
1 & 0.2008 & 0.2020 & 0.0745 & -0.0553 \\
0.2008 & 1 & 0.9821 & 0.0303 & -0.0901 \\
0.2020 & 0.9821 & 1 & 0.1106 & 0.0153 \\
0.0745 & 0.0303 & 0.1106 & 1 & 0.4231 \\
-0.0553 & -0.0901 & 0.0153 & 0.4231 & 1
\end{bmatrix}.
$$

The covariance matrix is further decomposed to derive the upper triangular matrix $L^T$ via Cholesky decomposition:

$$
L^T = \begin{bmatrix}
1 & 0.2008 & 0.2020 & 0.0745 & -0.0553 \\
0 & 0.9796 & 0.9611 & 0.0157 & -0.0806 \\
0 & 0 & 0.1885 & 0.4268 & 0.5518 \\
0 & 0 & 0 & 0.9011 & 0.2142 \\
0 & 0 & 0 & 0 & 0.8001
\end{bmatrix}.
$$
By multiplying \( \mathbf{Z} \) with \( \mathbf{L}^T \) (Equation (6.13)), the independent standard normal random samples are transformed into correlated standard normal random samples, i.e., \( \tilde{\mathbf{Z}}^c = \tilde{\mathbf{Z}} \cdot \mathbf{L}^T = [\tilde{Z}_{\Delta x}^c, \tilde{Z}_{v_f}^c, \tilde{Z}_{v_l}^c, \tilde{Z}_{acc_f}^c, \tilde{Z}_{acc_l}^c] \).

The final transformation is to apply first the CDF of the standard normal distribution (i.e., \( \Phi(\cdot) \)) to \( \tilde{\mathbf{Z}}^c \) to derive \( \Phi(\tilde{Z}_{\Delta x}^c), \Phi(\tilde{Z}_{v_f}^c), \Phi(\tilde{Z}_{v_l}^c), \Phi(\tilde{X}_{acc_f}^c), \) and \( \Phi(\tilde{Z}_{acc_l}^c) \). Then the inverse CDFs of the 5 kinematic inputs (see Figure 6.2) are used to produce the desired dependent random samples. For example, if \( \Phi(\tilde{Z}_{acc_f}^c) \) is 0.25, then according to the inverse CDF in Figure 6.2(d), the corresponding random sample of \( \tilde{X}_{acc_f} \) is 0.

Figure 6.2: Inverse CDFs of \( \Delta x, v_f, v_l, \text{acc}_f, \) and \( \text{acc}_l \) obtained from the empirical data.

For demonstration purposes, two experiments with two different sample sizes are conducted in this case study. The first experiment has a sample size of 1,024 for each kinematic input and uses the Sobol’ quasi-random sequence. It represents the scenario in which the sampling process can only afford a small number of runs of computationally expensive models. The second experiment has a sample size of 10,000 using Monte Carlo random sampling. This
experiment represents the sampling process for the computationally cheap models, in which a large number of model runs are affordable. The marginal distributions, scatter plots and the SRCCs of the random samples generated in these two experiments are shown in Figures 6.3 and 6.4, respectively. The corresponding LCCs and SRCCs are reported in Table 6.2.

Table 6.2: Correlation coefficients from the two experiments with sample size 1,024 and 10,000.

<table>
<thead>
<tr>
<th>Input Pairs</th>
<th>Sample Size = 1,024</th>
<th></th>
<th>Sample Size = 10,000</th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>LCC</td>
<td>SRCC</td>
<td>LCC</td>
<td>SRCC</td>
</tr>
<tr>
<td>∆x &amp; v_f</td>
<td>0.1912</td>
<td>0.1899</td>
<td>0.1953</td>
<td>0.1895</td>
</tr>
<tr>
<td>∆x &amp; v_l</td>
<td>0.1896</td>
<td>0.1912</td>
<td>0.1992</td>
<td>0.1915</td>
</tr>
<tr>
<td>∆x &amp; acc_f</td>
<td>0.0603</td>
<td>0.0673</td>
<td>0.0705</td>
<td>0.0731</td>
</tr>
<tr>
<td>v_f &amp; v_l</td>
<td>-0.0695</td>
<td>-0.0603</td>
<td>-0.0456</td>
<td>-0.0586</td>
</tr>
<tr>
<td>v_f &amp; acc_f</td>
<td>0.09763</td>
<td>0.9795</td>
<td>0.9768</td>
<td>0.9795</td>
</tr>
<tr>
<td>v_l &amp; acc_f</td>
<td>0.0140</td>
<td>0.0233</td>
<td>0.0263</td>
<td>0.0296</td>
</tr>
<tr>
<td>v_l &amp; acc_f</td>
<td>-0.1006</td>
<td>-0.0917</td>
<td>-0.0832</td>
<td>-0.0816</td>
</tr>
<tr>
<td>acc_f &amp; acc_l</td>
<td>0.0017</td>
<td>0.0092</td>
<td>0.0159</td>
<td>0.0183</td>
</tr>
</tbody>
</table>

A visual cross-comparison between Figures 6.1 and 6.4 shows that the marginal distributions in the experiment with 10,000 random samples are very similar to those of the empirical data. The main reason behind is this experiment generates a big number of random samples under the given marginal distributions, hence it results in a better coverage of the input space. However, it should be noticed that it might not always be feasible to run computationally expensive models with a big number of random samples. Therefore, it is also meaningful to check the performance of this sampling approach using a small sample size. Figure 6.3 shows that the marginal distributions in the experiment based on 1,024 quasi-random samples also present certain similarities with those of the empirical data. This finding indicates that even with a small sample size based on low discrepancy sequences (e.g., Sobol’ sequence), the proposed sampling approach can reach a satisfactory exploration of the input space.

Moreover, a KS two-sample test (see Equation (2.22)) is introduced to investigate if the marginal distributions of the random samples are significantly different with those of the empirical data\(^3\). The minimum $p$-values and the corresponding kinematic inputs from both experiments are reported in Table 6.3. It is found that both experiments yield rather high $p$-values (e.g., the minimum $p$-value from Experiment 1 is 0.8234 for input $v_f$). These results further prove that for each individual kinematic input, the marginal distributions of random samples generated in both experiments are consistent with that of the empirical data.

Table 6.3 also reports the maximum absolute differences between the empirical data and the random samples in the two experiments with respect to LCCs and SRCCs. Overall, the maximum absolute differences are rather small, especially for SRCC. Hence, it can be inferred

---

\(^3\) If the $p$-value obtained from a KS two-sample test is small, it means that the two groups of data have different distributions.
Figure 6.3: Marginal distribution, scatter plot and the SRCC of the five kinematic inputs based on 1,024 random samples. Plots in the diagonal: marginal distribution of the corresponding input. Plots below the diagonal: scatter plots of the two inputs located in the corresponding row and column. Plots above the diagonal: SRCC of the two inputs located in the corresponding row and column.
Figure 6.4: Marginal distribution, scatter plot and the SRCC of the five kinematic inputs based on 10,000 random samples. Plots in the diagonal: marginal distribution of the corresponding input. Plots below the diagonal: scatter plots of the two inputs located in the corresponding row and column. Plots above the diagonal: SRCC of the two inputs located in the corresponding row and column.
that the dependence structures of the random samples in the two experiments are also quite similar to that of the empirical data. In addition, by making a pair-wise comparison of the correlation coefficients of the empirical data and the random samples in Tables 6.1 and 6.2, it is found that the orders of both LCCs and SRCCs for all input pairs in the two experiments are exactly the same as those of the empirical data. For instance, \( v_f - v_l \) has the strongest positive correlation, followed by \( \text{acc}_f - \text{acc}_l \), \( \Delta x - v_f \) and \( \Delta x - v_l \). Therefore, the accuracy of the random samples generated in both experiments are acceptable, and the random samples can be used for the extended variance-based SA and the extended EE method for analyzing the sensitivity information of the dependent kinematic inputs (see the case studies in Sections 7.3 and 8.3).

Table 6.3: Comparison of the random samples from the two experiments.

<table>
<thead>
<tr>
<th></th>
<th>Experiment 1</th>
<th>Experiment 2</th>
</tr>
</thead>
<tbody>
<tr>
<td>Sample size</td>
<td>1,024</td>
<td>10,000</td>
</tr>
<tr>
<td>Min. ( p )-value by KS test</td>
<td>0.8234 ((v_f))</td>
<td>0.8894 ((\text{acc}_f))</td>
</tr>
<tr>
<td>Maximum Absolute Difference for LCC</td>
<td>0.093 ((\text{acc}_f &amp; \text{acc}_l))</td>
<td>0.076 ((\Delta x &amp; \text{acc}_f))</td>
</tr>
<tr>
<td>Maximum Absolute Difference for SRCC</td>
<td>0.0074 ((\Delta x &amp; \text{acc}_l))</td>
<td>0.0057 ((\Delta x &amp; \text{acc}_l))</td>
</tr>
</tbody>
</table>

### 6.4 Summary of the Chapter

This chapter presents a general approach for generating random samples of dependent parameters. It utilizes the Gaussian copula in the sampling process, which makes it attractive for generating random samples from any given arbitrary marginal distribution. Furthermore, the rank correlation SRCC is employed instead of the commonly adopted linear correlation LCC in the sampling process, which helps to retain the given dependence structure throughout the non-linear transformation of the Gaussian copula (i.e., the inverse Nataf transformation).

A case study for generating random samples of the 5 kinematic inputs of the Wiedemann-74 car-following model is included to demonstrate the application of this approach. It has shown that the marginal distributions and correlation coefficients of the random samples are comparable with those obtained from the empirical car-following data. In particular, the 1,024 random samples, which are generated by employing the Sobol’ quasi-random sequence in the sampling process, also present consistent marginal distributions and SRCC as the empirical data. This suggests that the proposed sampling approach can be used to generate random samples for computationally expensive models, for which a large number of model runs is not always affordable.

As a summary, the copula-based sampling approach illustrated in this chapter is a flexible and efficient approach for generating dependent random samples. These random samples can be further used in both qualitative and quantitative estimation of the sensitivity information of dependent parameters. The details are introduced in Chapters 7 and 8, respectively.
Chapter 7

Quantitative Sensitivity Analysis for Model with Dependent Parameters

This chapter contains the findings from the following research papers:


7.1 Introduction

The discussion from Chapter 6 shows that when dependence exists among the model parameters, a proper SA including the sampling process should not only consider the marginal distributions of individual parameters, but also take the global distribution of all parameters into account. Thus, the focus of this chapter is to introduce a global method for quantifying the sensitivity of dependent model parameters. To this end, this chapter proposes a quantitative SA based on dependent random samples. Specifically, the copula-based sampling approach introduced in Chapter 6 is used first to generate conditional and unconditional dependent random samples under given marginal distributions and dependence structure. Then the extended Sobol’ formulas (Kucherenko et al., 2012; Mara et al., 2015) and the QMC estimator are employed for estimating the sensitivity indexes for dependent parameters. The major advantage of this method is that it can deal with models containing both independent and dependent parameters with different arbitrary distributions.

The application of the proposed sampling-based SA is illustrated with the Wiedemann-74 car-following model with both dependent and independent factors. Results show that this method is able to accurately quantify the sensitivity for all factors. Thus, as a general global SA method, it can be transferred like a standard quantitative SA tool for any traffic simulation model or complex model in the wider scientific community, especially when dependent parameters exist.

The rest of this chapter is organized as follows. Section 7.2 presents the methodology of the proposed SA method, including the variance decomposition for dependent parameters, and the analytical formulation and QMC estimation of the sensitivity indexes. Section 7.3 employs a case study based on the Wiedemann-74 car-following model to illustrate the application of the proposed SA method. Section 7.4 summarizes the findings in this chapter.

7.2 Methodology

This section illustrates the methodology of the proposed SA method for models with dependent parameters. This is relevant as no such method can be found in the existing literature from the transportation research community, despite the fact that many traffic models contain dependent parameters. The SA method discussed in this section is based on the variance decomposition of the dependent parameters, and it uses the QMC simulations for estimating the sensitivity indexes. The copula-based sampling approach introduced in Chapter 6 is employed here for generating dependent random samples. The details are presented below.

7.2.1 Variance decomposition for dependent parameters

When the model contains dependent parameters, the ANOVA decomposition presented in Equation (2.12) can not be simply applied without considering the joint distributions of all parameters (Da Veiga et al., 2009). In fact, according to Zhang et al. (2015a), Kucherenko et al. (2012), and Mara et al. (2015), the total variance contributed by a parameter includes three parts: the main variance, the variance caused by interactions, and the variance caused by dependence. Several studies have discussed the variance decomposition and the corresponding
sensitivity indexes for models with dependent parameters (see e.g., Xu and Gertner, 2008; Li et al., 2010; Kücherenko et al., 2012; Mara et al., 2015; Zhang et al., 2015a). This section recalls the findings presented in Kücherenko et al. (2012) and Mara et al. (2015), in which the non-parametric methods are used to decompose the variances and compute the sensitivity indexes.

Considering a model with \( k \) parameters, i.e., \( \mathbf{X} = \{X_1, X_2, \cdots, X_k\} \). Let \( \mathbf{X}_{-i} \) be a set containing all model parameters except \( X_i (i \in [1, k]) \). In Kücherenko et al. (2012), two types of variances are introduced:

- \( V^\text{full}_i \): the dependent main variance of \( X_i \), which includes the main variance of \( X_i \), and the variance caused by the dependence of \( X_i \) with \( \mathbf{X}_{-i} \), but excludes the variance due to the interaction of \( X_i \) with \( \mathbf{X}_{-i} \); and
- \( V^\text{ind}_i \): the independent total variance of \( X_i \), which includes the main variance of \( X_i \), and the variance due to the interaction of \( X_i \) with \( \mathbf{X}_{-i} \), but excludes the variance caused by the dependence of \( X_i \) with \( \mathbf{X}_{-i} \).

In a recent research, Mara et al. (2015) proposed two additional variances:

- \( V^\text{ind}_i \): the independent main variance of \( X_i \), which only includes the main variance of \( X_i \), but excludes the variance caused by the interaction and dependence of \( X_i \) with \( \mathbf{X}_{-i} \); and
- \( V^\text{full}_i \): the dependent total variance of \( X_i \), which includes the main variance of \( X_i \), and the variance caused by the interaction and dependence of \( X_i \) with \( \mathbf{X}_{-i} \).

According to the law of total variance, \( V(Y) = V^\text{full}_i + V^\text{ind}_i \). Let \( V^\text{full}_i = V^\text{ind}_i + V^\text{total}_i \) with \( \mathbf{X}_{-i} \).

The four variances mentioned above can be further normalized with \( V(Y) \), and accordingly four sensitivity indexes can be obtained:

\[
S^\text{ind}_i = \frac{V^\text{ind}_i}{V(Y)}, \quad \text{(7.1a)}
\]
\[
S^\text{full}_i = \frac{V^\text{full}_i}{V(Y)}, \quad \text{(7.1b)}
\]
\[
S^\text{ind}_i = \frac{V^\text{ind}_i}{V(Y)}, \quad \text{(7.1c)}
\]
\[
S^\text{full}_i = \frac{V^\text{full}_i}{V(Y)}. \quad \text{(7.1d)}
\]

\( S^\text{ind}_i \) and \( S^\text{ind}_i \) are called independent sensitivity indexes (Mara et al., 2015), and \( S^\text{full}_i \) and \( S^\text{full}_i \) are known as full sensitivity indexes (Mara and Tarantola, 2012). Similar as the Sobol’ sensitivity indexes for independent parameters, the inequalities \( S^\text{ind}_i \leq S^\text{ind}_i \) and \( S^\text{full}_i \leq S^\text{full}_i \) also hold here for dependent parameters. However, because the impacts of dependence can be either positive or negative, it is not necessarily true that \( S^\text{ind}_i \leq S^\text{full}_i \).

\[\text{In Mara and Tarantola (2012), the full sensitivity indexes are denoted as } S_i \text{ and } S_i^\text{full}. \text{ In order to avoid the confusion with the sensitivity indexes for independent parameters (Equations (2.16a) and (2.16b)), } S_i^\text{full} \text{ and } S_i^\text{full} \text{ are used to represent the full sensitivity indexes in this dissertation.}\]
According to Kucherenko et al. (2012), when there is no dependence impact, i.e., when the parameters are independent parameters, the variance of the model are:

\[ S_{i,\text{full}} \leq S_{i,\text{ind}}, \text{ or } S_{i,\text{ind}} \leq S_{i,\text{full}} \]  

(see the example in Kucherenko et al., 2012). In addition, when there is no dependence impact, i.e., when the parameters are independent parameters, the independent sensitivity indexes have the same values as the corresponding full sensitivity indexes.

Moreover, when \( S_{i,\text{ind}} \approx 0 \) but \( S_{i,\text{full}} \gg 0 \), it can be inferred that the influence of \( X_i \) on the output variance is only due to its dependence with other parameters. On the contrary, if \( S_{i,\text{ind}} \gg 0 \) it means that almost all variance contributions of \( X_i \) are from \( X_i \) alone and/or its interactions with other parameters. In both cases, \( X_i \) is identified as an influential parameter since it has contributions either through its dependence with other influential parameters (i.e., the former case), or by itself and/or the interactions with other parameters (i.e., the latter case). \( X_i \) can only be classified as a non-influential parameter when both \( S_{i,\text{ind}} \) and \( S_{i,\text{full}} \) are close to 0. On the other hand, it can be inferred that \( X_i \) will surely be an influential parameter when both \( S_{i,\text{ind}} \gg 0 \) and \( S_{i,\text{full}} \gg 0 \) hold, i.e., no matter if the dependence influence is positive or negative, \( X_i \) will have strong impact on the variance of the model output due to its main variance contribution.

### 7.2.2 Analytical formulation of sensitivity indexes

According to Kucherenko et al. (2012) and Mara et al. (2015), \( S_{i,\text{ind}} \) and \( S_{i,\text{full}} \) can be computed analytically or numerically. Consider a model \( f \) that is a square integrable function over the input space \( \Omega^k (\Omega^k \subset \mathbb{R}^k) \). Let \( p(X_i) \) be the marginal PDF of parameter \( X_i \), and \( p(X) \) be the joint PDF of all parameters, i.e., \( p(X) = p(X_1, \cdots, X_k) \). The expectation and variance of the model are:

\[
E(Y) = \int_{\Omega^k} f(X)p(X) \, dX = f_0,  
\]

\[
V(Y) = \int_{\Omega^k} f^2(X)p(X) \, dX - f_0^2.  
\]

The analytical formulas to compute the independent and full sensitivity indexes are presented below (for the development of these formulas see Kucherenko et al., 2012; Mara et al., 2015):

\[
S_{i,\text{ind}} = \frac{1}{V(Y)} \left[ \int_{\Omega^k} \left[ \int_{\Omega^{k-1}} f(\bar{X}_i, X_{-i})p(X_{-i}) \, dX_{-i} \right]^2 p(\bar{X}_i|X_{-i}) \, d\bar{X}_i - \left[ \int_{\Omega} \left[ \int_{\Omega^{k-1}} f(\bar{X}_i, X_{-i})p(X_{-i}) \, dX_{-i} \right] \cdot p(\bar{X}_i|X_{-i}) \, d\bar{X}_i \right]^2 \right],  
\]

\[
S_{i,\text{full}} = \frac{1}{V(Y)} \left[ \int_{\Omega} \left[ \int_{\Omega^{k-1}} f(X_i, \bar{X}_{-i})p(\bar{X}_{-i}|X_i) \, d\bar{X}_{-i} \right]^2 p(X_i) \, dX_i - \left[ \int_{\Omega} \left[ \int_{\Omega^{k-1}} f(X_i, \bar{X}_{-i})p(\bar{X}_{-i}|X_i) \, d\bar{X}_{-i} \right] \cdot p(X_i) \, dX_i \right]^2 \right].  
\]
where the model is a black-box (a common case for many commercial simulation models), it is also not possible to compute the sensitivity indexes analytically.

A feasible solution in practice is to estimate the sensitivity indexes in a Monte Carlo (MC) framework. To this end, the following MC estimators are used (more details are given in Kucherenko et al., 2012; Mara et al., 2015):

\[
\hat{V}(Y) = \frac{1}{N} \sum_{r=1}^{N} \left[ f(x_{i,r}, x_{-i,r}) \right]^2 - \frac{1}{N} \sum_{r=1}^{N} f(x_{i,r}, x_{-i,r})^2, \tag{7.5a}
\]

\[
\hat{S}_{i}^{\text{ind}} = \frac{1}{N \cdot \hat{V}(Y)} \sum_{r=1}^{N} \left[ f(\tilde{x}_{i,r}, x_{-i,r}) - f(x_{i,r}, x_{-i,r}) \right] f(x_{i,r}, x_{-i,r}), \tag{7.5b}
\]

\[
\hat{S}_{i}^{\text{full}} = \frac{1}{N \cdot \hat{V}(Y)} \sum_{r=1}^{N} \left[ f(x_{i,r}, \tilde{x}_{-i,r}) - f(x_{i,r}, x_{-i,r}) \right] f(x_{i,r}, x_{-i,r}), \tag{7.5c}
\]

\[
\tilde{S}_{i}^{\text{ind}} = \frac{1}{2N \cdot \hat{V}(Y)} \sum_{r=1}^{N} \left[ f(x_{i,r}, x_{-i,r}) - f(\tilde{x}_{i,r}, x_{-i,r}) \right]^2, \tag{7.5d}
\]

\[
\tilde{S}_{i}^{\text{full}} = \frac{1}{2N \cdot \hat{V}(Y)} \sum_{r=1}^{N} \left[ f(x_{i,r}, x_{-i,r}) - f(x'_{i,r}, \tilde{x}_{-i,r}) \right]^2, \tag{7.5e}
\]

where \( N \) is the number of MC simulations. \( \{x_{i,r}, x_{-i,r}\} \) with \( r \in [1, N] \) is the \( r \)-th random sample generated from the joint PDF \( p(X) \). \( \{x'_{i,r}, x'_{-i,r}\} \) is a different set of random samples generated from the same joint PDF \( p(X) \), i.e., \( x_{i,r} \) and \( x'_{i,r} \) are independent of each other but
they follow the same marginal PDF of \( X_i \). \( \{ x'_{i,r}, x_{-i,r} \} \) is the \( r \)-th random sample generated based on a given \( x_{-i,r} \), i.e., it is generated from the conditional PDF \( p(\hat{x}_i|X_{-i}) \). \( \{ \hat{x}'_{i,r}, \hat{x}_{-i,r} \} \) is the \( r \)-th random sample generated based on a given \( x'_{i,r} \), i.e., it is generated from the conditional PDF \( p(X_{-i}|x_i) \).

The unconditional dependent samples (i.e., \( \{ x_{i,r}, x_{-i,r} \} \) and \( \{ x'_{i,r}, x'_{-i,r} \} \)) and the conditional dependent samples (i.e., \( \{ x'_{i,r}, x_{-i,r} \} \) and \( \{ x'_{i,r}, \hat{x}_{-i,r} \} \)) can be generated using the copula-based sampling approach introduced in Chapter 6 in combination with a quasi-random sequence.

The quasi-random sequence employed here is the Sobol’ sequence (Sobol’, 1976) due to its low discrepancy and high efficiency (see Section 2.2.5). An \( N \)-by-\( 2k \) matrix of Sobol’ random numbers can be generated:

\[
U = \begin{bmatrix}
  u_{1,1} & \ldots & u_{1,k} \\
  u_{1,2} & \ldots & u_{1,k+1} \\
  \vdots & \ddots & \vdots \\
  u_{1,N} & \ldots & u_{1,N+k-1}
\end{bmatrix} = \langle A|B \rangle, \tag{7.6}
\]

where \( A = \begin{bmatrix}
  u_{1,1} & \ldots & u_{1,k} \\
  u_{1,2} & \ldots & u_{1,k+1} \\
  \vdots & \ddots & \vdots \\
  u_{1,N} & \ldots & u_{1,N+k-1}
\end{bmatrix}, B = \begin{bmatrix}
  u_{k+1,1} & \ldots & u_{2k,1} \\
  u_{k+1,2} & \ldots & u_{2k,2} \\
  \vdots & \ddots & \vdots \\
  u_{k+1,N} & \ldots & u_{2k,N}
\end{bmatrix} \).

According to the characteristics of the Sobol’ sequence, each column vector in \( U \) is independent of the other column vectors, and the elements in any column are uniformly distributed in \( U[0, 1] \). Thus, the unconditional dependent random sample vector \( \{ x_{i,r}, x_{-i,r} \} \) can be generated through transforming the \( r \)-th row in \( A \), i.e., \( [u_{1,r}, u_{2,r}, \ldots, u_{k,r}] \), using the IC procedure introduced in Section 6.2.3; \( \{ x'_{i,r}, x'_{-i,r} \} \) is generated in the same way but using the \( r \)-th row in \( B \). The conditional vectors \( \{ x_{i,r}, x'_{i,r} \} \) and \( \{ x_{i,r}, x'_{-i,r} \} \) are not so straightforward to generate. The process for generating the conditional vector \( \{ x'_{i,r}, \hat{x}_{-i,r} \} \) is described below. The conditional vector \( \{ \hat{x}'_{i,r}, \hat{x}_{-i,r} \} \) can be constructed similarly (see Kucherenko et al., 2012; for more details).

First, construct an independent random vector with standard normal distribution, i.e., \( [z_{i,r}, z_{-i,r}] = [z_{i,r}, z_{i+1,r}, \ldots, z_{k,r}, z_{i+1,r}, \ldots, z_{i-1,r}] \), using the independent random vector \( [u_{i,r}, u_{i+1,r}, \ldots, u_{k,r}, u_{i,r}, \ldots, u_{i-1,r}] \). This can be done by first shifting the corresponding elements in the \( r \)-th row of \( A \), and then applying the inverse CDF of standard normal distribution \( \Phi^{-1}(\cdot) \) to each element in the new vector. Let \( [z'_{1,r}, \ldots, z'_{k,r}] \) be a dependent random vector generated from the independent random vector \( [u_{k+1,r}, \ldots, u_{2k,r}] \) in \( B \) (see Step 5 of the IC procedure in Section 6.2.3). Since \( A \) and \( B \) are independent of each other, the \( i \)-th element of \( [z'_{1,r}, \ldots, z'_{k,r}] \), i.e., \( z'_{i,r} \), is also independent of \( z_{-i,r} \). The covariance matrix with respect to \( \{ X_i, X_{-i} \} \) (see Equation (6.11)), i.e., \( \Sigma_{z_i} \), can be derived as:
where

\[
\rho_{\alpha, \beta} = 2 \sin \left( \frac{\pi}{6} \rho_{\alpha, \beta} \right), \forall \alpha, \beta \in [1, k];
\]

\[
\Sigma_{i,-i} = \left[ \rho_{i,i+1}, \ldots, \rho_{i,k}, \rho_{i,1}, \ldots, \rho_{i,i-1} \right];
\]

\[
\Sigma_{-i,i} = \left[ \rho_{i+1,i}, \ldots, \rho_{k,i}, \rho_{i,i}, \ldots, \rho_{i-1,i} \right]^T;
\]

and

\[
\Sigma_{z_i} = \begin{bmatrix}
1 & \rho_{i,i+1} & \cdots & \rho_{i,k} & \rho_{i,1} & \cdots & \rho_{i,i-1} \\
\rho_{i+1,i} & 1 & \cdots & \rho_{i,k} & \rho_{i,1} & \cdots & \rho_{i,i-1} \\
\vdots & \vdots & \ddots & \vdots & \vdots & \ddots & \vdots \\
\rho_{k,i} & \rho_{k,i+1} & \cdots & 1 & \rho_{k,1} & \cdots & \rho_{k,i-1} \\
\rho_{i,i+1} & \rho_{i+1,i} & \cdots & \rho_{i,k} & 1 & \cdots & \rho_{i,i-1} \\
\vdots & \vdots & \ddots & \vdots & \vdots & \ddots & \vdots \\
\rho_{i-1,i} & \rho_{i-1,i+1} & \cdots & \rho_{i-1,k} & \rho_{i-1,1} & \cdots & 1 \\
\end{bmatrix}
\]

\[
\Sigma_{z_i} = \begin{bmatrix}
1 & \Sigma_{i,i} \\
\Sigma_{-i,i} & \Sigma_{-i,-i} \\
\end{bmatrix}, \quad (7.7)
\]

In the next step, let \( \Sigma_{\tilde{z}_i} \) be the covariance matrix of the dependent conditional vector \( \tilde{z}^c_i \), \( \tilde{z}^c_i = \left[ \tilde{z}^c_{i,i+1}, \ldots, \tilde{z}^c_{i,k}, \tilde{z}^c_{i,1}, \ldots, \tilde{z}^c_{i,i-1} \right] \). It is derived as:

\[
\Sigma_{\tilde{z}_i} = \Sigma_{-i,i} - \Sigma_{i,-i} \cdot \Sigma_{-i,i}.
\]

Let \( L_{\tilde{z}_i} \) be the lower triangular matrix derived by the Cholesky decomposition of \( \Sigma_{\tilde{z}_i} \), so that \( \Sigma_{\tilde{z}_i} = L_{\tilde{z}_i} \cdot L_{\tilde{z}_i}^T \). The dependent samples for the conditional vector \( \tilde{z}^c_i \), i.e., \( \tilde{z}^c_{i,i} = \left[ \tilde{z}^c_{i,i+1}, \ldots, \tilde{z}^c_{i,k}, \tilde{z}^c_{i,1}, \ldots, \tilde{z}^c_{i,i-1} \right] \), can be obtained as:

\[
\tilde{z}^c_{i,i} = z_{-i,i} \cdot L_{\tilde{z}_i}^T + z_{i,i}^c \cdot \Sigma_{i,-i}.
\]

Finally, the base vector \( \left[ \tilde{z}^c_{i,i+1}, \ldots, \tilde{z}^c_{i,k}, \tilde{z}^c_{i,1}, \ldots, \tilde{z}^c_{i,i-1} \right] \) can be formed by combining \( z_{-i,i}^c \) and \( \tilde{z}^c_{i,i} \). Then \( \{ x'_{i,i}, \tilde{x}_{i,i} \} \) can be achieved by applying the component-wise transformation \( T_i(\cdot) \) (see Equation (6.6)) to the base vector.

The model will then be evaluated using all dependent samples for \( \{ x_{i,i}, x_{i,i} \}, \{ x'_{i,i}, x'_{i,i} \}, \{ \tilde{x}_{i,i}, \tilde{x}_{i,i} \}, \) and \( \{ x'_{i,i}, \tilde{x}_{i,i} \} \), so as to produce the outputs \( f(x_{i,i}, x_{i,i}), f(x'_{i,i}, x'_{i,i}), f(\tilde{x}_{i,i}, x_{i,i}), \) and \( f(x'_{i,i}, \tilde{x}_{i,i}). \) The independent and full sensitivity indexes are computed using Equations (7.5b) to (7.5e) accordingly. The computation of the independent and full sensitivity indexes of \( X_i \) requires \( N \) runs of the model based on the unconditional dependent samples \( \{ x_{i,i}, x_{i,i} \} \) and \( \{ x'_{i,i}, x'_{i,i} \} \), as well as the conditional dependent samples \( \{ \tilde{x}_{i,i}, \tilde{x}_{i,i} \} \) and
\{x_{ij}, \bar{x}_{ij}\}}, respectively. Thus, the total model runs required to compute the independent and full sensitivity indexes for all \(k\) parameters is \(4kN\). In general, \(N\) needs to be a big number (e.g., from \(10^3\) to \(10^5\)) for the QMC estimator to get accurate results.

### 7.3 Case Study

In this section a case study which employs the Wiedemann-74 car-following model is used to illustrate the methodology and outcome of the proposed SA approach for dependent parameters. In addition, to demonstrate the potential consequences of not taking the parameter dependency into account in the SA, a hypothetical scenario in which all dependent parameters are assumed to be independent is included. Note that the proposed SA is a general approach, and the steps described above will not change if different models, data distributions, or dependence structures are used.

#### 7.3.1 Experimental design

The introduction of the Wiedemann-74 model can be found in Section 5.3.2. Due to the limitation of the sequential SA in dealing with dependent parameters, the case study presented in Section 5.3.2 groups the 7 dependent kinematic inputs, i.e., \(x_f, x_l, v_f, v_l, acc_f, acc_l\) and \(L_l\), as a single factor \(Kin\) in the SA. It is found that \(Kin\) is the most influential factor, which accounts over 50% and 85% of the main and total variances respectively. In order to check the impacts of individual kinematic inputs, it is necessary to perform a proper SA that correctly maintains the sensitivity information of the dependent parameters. Thus, the SA approach proposed in this chapter is used to quantify the sensitivity of individual kinematic inputs. Moreover, as already explained in Section 6.3, to avoid invalid samples in cases where \(x_f - L_l < x_l, \Delta x = x_l - x_f - L_l\) is included to replace the original kinematic inputs \(x_l, x_f, \) and \(L_l\) in the SA. In total 5 kinematic inputs, i.e., \(\Delta x, v_f, v_l, acc_f, \) and \(acc_l\) are analyzed in this case study. Such setting is the same as that for the case study for generating dependent random samples in Section 6.3. The distributions and corresponding correlation coefficients for these 5 kinematic inputs are already reported in Figure 6.1 and Table 6.1.

In addition to the 5 kinematic inputs, 12 model parameters are selected from the 24 parameters in the SA, i.e., \(BX_1, BX_2, EX_1, EX_2, CX_0, BMAX_1, BMIN_1, BNULL_1, VDES, R_1, R_2, \) and \(NR\). This is due to the screening results of the quasi-OTEE analysis in the sequential SA in Chapter 5, in which these parameters are identified as potentially influential parameters. To cross-compare with the findings in Section 5.3.2, these parameters are assumed to be independent of each other and the 5 kinematic inputs, and they have the same distributions as those presented in Section 5.3.2. The remaining model parameters are non-influential, and they will be fixed to the default values in this case study. Overall, 17 factors are considered in the SA, including 9 independent factors with uniform distribution, 3 independent factors with normal distribution, and 5 dependent factors with empirical distributions (see Table 7.1).

Since there is no universal rule that defines a proper sample size in the QMC estimation, the sample size can vary e.g., from some hundreds to several thousands. Here, the sample size is arbitrarily set to 1,024 (i.e., \(2^{10}\)) in order to achieve the lowest discrepancy of the Sobol’ sequence (Sobol’, 1976). To estimate the robustness of the SA results, the proposed
Table 7.1: 17 factors analyzed in the sensitivity analysis of the Wiedemann-74 model.

<table>
<thead>
<tr>
<th>#</th>
<th>Name</th>
<th>Marginal Distribution</th>
<th>Dependence</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>BX₁</td>
<td>U[0.5, 4]</td>
<td>Independent</td>
</tr>
<tr>
<td>2</td>
<td>BX₂</td>
<td>U[0.01, 5]</td>
<td>Independent</td>
</tr>
<tr>
<td>3</td>
<td>EX₁</td>
<td>U[0.5, 10]</td>
<td>Independent</td>
</tr>
<tr>
<td>4</td>
<td>EX₂</td>
<td>U[0.01, 10]</td>
<td>Independent</td>
</tr>
<tr>
<td>5</td>
<td>CX₀</td>
<td>U[0.1, 50]</td>
<td>Independent</td>
</tr>
<tr>
<td>6</td>
<td>BMAX₁</td>
<td>U[0.1, 0.2]</td>
<td>Independent</td>
</tr>
<tr>
<td>7</td>
<td>BMIN₁</td>
<td>U[0.1, 8]</td>
<td>Independent</td>
</tr>
<tr>
<td>8</td>
<td>BNULL₁</td>
<td>U[0.01, 10]</td>
<td>Independent</td>
</tr>
<tr>
<td>9</td>
<td>VDES</td>
<td>U[10, 20]</td>
<td>Independent</td>
</tr>
<tr>
<td>10</td>
<td>R₁</td>
<td>N(0.5, 0.15)</td>
<td>Independent</td>
</tr>
<tr>
<td>11</td>
<td>R₂</td>
<td>N(0.5, 0.15)</td>
<td>Independent</td>
</tr>
<tr>
<td>12</td>
<td>NR</td>
<td>N(0.5, 0.15)</td>
<td>Independent</td>
</tr>
<tr>
<td>13</td>
<td>Δx</td>
<td>Empirical distribution</td>
<td>Dependent</td>
</tr>
<tr>
<td>14</td>
<td>vᵢ</td>
<td>Empirical distribution</td>
<td>Dependent</td>
</tr>
<tr>
<td>15</td>
<td>vᵣ</td>
<td>Empirical distribution</td>
<td>Dependent</td>
</tr>
<tr>
<td>16</td>
<td>accᵢ</td>
<td>Empirical distribution</td>
<td>Dependent</td>
</tr>
<tr>
<td>17</td>
<td>accᵣ</td>
<td>Empirical distribution</td>
<td>Dependent</td>
</tr>
</tbody>
</table>

U[a, b]: uniform distribution with lower bound a and upper bound b.  
N(μ, σ): normal distribution with mean μ and standard deviation σ.

SA is performed 1,000 times via bootstrapping (Efron and Tibshirani, 1993). In other words, the 1,024 quasi-random samples will be resampled 1,000 times with replacement, and the computation of the sensitivity indexes will be repeated 1,000 times with these samples accordingly. The final results will be the mean and error bounds of the corresponding sensitivity indexes based on the 1,000 QMC estimations.

Moreover, for demonstration purposes, the same SA is performed in a hypothetical case, in which the 5 dependent kinematic inputs are considered as independent (this can be achieved by setting the relevant correlation coefficients to zero). A cross-comparison of SA results can therefore be made to show the consequences when the parameter dependence is not correctly modeled in the SA.

### 7.3.2 Results of the case study

The SA results in the case where the dependence structure of the 5 kinematic inputs is correctly modeled are reported in Figure 7.1. Specifically, Figure 7.1(a) presents the estimated independent sensitivity indexes, and Figure 7.1(b) presents the estimated full sensitivity indexes for the 17 factors. The bars are the mean values of the corresponding independent sensitivity indexes (i.e., $S_{ind}$ and $S^{T,ind}$ in Figure 7.1(a)) and full sensitivity indexes (i.e., $S_{full}$ and $S^{T,full}$ in Figure 7.1(b)) from the 1,000 QMC estimations with bootstrapping. The error bars represent the 5% and 95% percentiles of the corresponding sensitivity indexes, i.e., the
90% BCIs of the estimation. Overall, one can see that all BCIs are narrow, which indicates that the sample size in QMC is adequate and the SA results are sufficiently stable.

Figure 7.1: Estimated independent and full sensitivity indexes for the 17 factors of Wiedemann-74 car-following model in the real case (dependent kinematic inputs have the same correlations as empirical data). In both figures, bars are the mean values, and error bars correspond to the bounds of 90% BCIs of the estimation.

(a) Independent sensitivity indexes for the 17 factors (dependent kinematic inputs).

(b) Full sensitivity indexes for the 17 factors (dependent kinematic inputs).

It is worth mentioning that the independent and full sensitivity indexes are general terms from SA research for quantifying the impacts of the variance of model parameters on the variance of model output. As an example, in both Figures 7.1(a) and 7.1(b), all sensitivity indexes of $v_l$ are higher than those of $acc_l$. This does not necessarily suggest the follower reacts slower to the change of acceleration than to the change of speed of the leader. In fact, it means that the variations in the follower’s acceleration are expected to be higher as a result of some variations in $v_l$ than a result of the same variations in $acc_l$. In other words, the variances contributed by $v_l$ itself are much stronger than the main variances in $acc_l$. Moreover, since the 12 model parameters, i.e., $BX_1, BX_2, EX_1, EX_2, CX_0, BMAX_1, BMIN_1, BNULL_1, VDES, R_1, R_2, and NR$, are set to be independent in the SA, the independent sensitivity indexes presented in Figure 7.1(a) and the full sensitivity indexes presented in Figure 7.1(b) are the same for these factors.

A visual inspection of Figure 7.1 shows that the most influential factor is $\Delta x$. The reason is the two main indexes, i.e., $S^\text{ind}_{\Delta x}$ and $S^\text{full}_{\Delta x}$, are much bigger than those of the other factors. In other words, the variances contributed by $\Delta x$ itself are much stronger than the main variances
which indicate that most impacts of (i.e., interaction and dependence) with other factors. A further comparison between around 0.18) suggests that vehicle speeds rather than the variations of vehicle accelerations. Therefore, to reduce the variance of the other influential factors will be indirectly fixed, and it may lead to wrong variance contributions by. The least influential factors are (around 0.12). It means that although the main and higher order variance contributions by are quite limited, the impacts caused by the dependence of with other influential factors to the variance of the model output are not trivial. Thus, if treating as a non-influential factor and fixing its variance in the model calibration, the variance of the other influential factors will be indirectly fixed, and it may lead to wrong calibration results. For this reason should be included as an influential factor according to the SA results, although it is not as influential as the other 4 kinematic inputs.

The other influential factors are , , , and , which have the two total sensitivity indexes and below 0.05. Therefore these parameters can be fixed without influencing the variance of the model output. The factor has a very low but a high (around 0.12). It means that although the main and higher order variance contributions by are quite limited, the impacts caused by the dependence of with other influential factors to the variance of the model output are not trivial. Thus, if treating as a non-influential factor and fixing its variance in the model calibration, the variance of the other influential factors will be indirectly fixed, and it may lead to wrong calibration results. For this reason should be included as an influential factor according to the SA results, although it is not as influential as the other 4 kinematic inputs.

The other influential factors are , and , whose and are over 0.05. In summary, 13 factors, i.e., , , , , , , , , , and , are identified as influential factors. This indicates that no matter which combinations of the values of the kinematic inputs are used, an efficient calibration can be performed by first calibrating , then and , and so on for the other parameters. It also shows that for all combinations of model parameters, the variations of the model output are mainly due to the variations of vehicle gaps and vehicle speeds rather than the variations of vehicle accelerations. Therefore, to reduce the uncertainties when the Wiedemann-74 model is used for modeling the car-following process, the uncertainties in vehicle gaps and vehicle speeds need to be cautiously managed. Although it is unrealistic to completely eliminate such uncertainties (e.g., due to the heterogeneity of driver and vehicle characteristics), general practices such as adopting a more accurate tool for measuring vehicle trajectories, or using data obtained from long observation periods (see Punzo et al., 2015) are recommended in this regard. By the same token, uncertainties in vehicle acceleration, however, do not need to be especially addressed. Therefore, not much effort needs to be spent on increasing the accuracy for measuring vehicle acceleration compared to measuring the vehicle gaps and speeds.
When making cross-comparison of the sensitivity indexes of independent factors in Figures 5.8 and 7.1; BMI is always the most influential independent factor, and its sensitivity indexes are almost the same in both studies. Additionally, the independent factors $EX_1$, $BX_1$, $CX_0$, $BX_2$, $NR$, $R_1$ and $R_2$ are also found to be influential according to the results presented in Figure 5.8; although some small differences exist in their corresponding sensitivity indexes. Since the total sensitivity indexes of these factors are relatively small in both case studies (all between 0.05 and 0.1), and QMC may have some deficiencies for estimating small sensitivity indexes (Saltelli, 2002), it can still be inferred that the sensitivity indexes derived by the proposed SA in this chapter for independent factors are consistent with those obtained in the case study in Section 5.3.2. These findings demonstrate that the proposed SA method is able to correctly derive the sensitivity indexes not only for independent factors but also for dependent factors with arbitrary distributions. Thus, compared to other SA methods in the existing literature or in practices within the transportation research community, which are mainly applicable to models with independent factors, this method is obviously more advanced in terms of problem solving ability.

Figure 7.2 presents the SA results for the hypothetical case, in which the 5 kinematic inputs still follow the marginal distributions of the empirical data, but they are assumed to be independent in the QMC estimation. Recall that when a factor is independent of all other factors, it has the same independent and full sensitivity indexes. For this reason the sensitivity indexes reported in Figures 7.2(a) and 7.2(b) are actually the same for the hypothetical case. A visual inspection shows the SA results in Figure 7.2 are quite different to those presented in Figure 7.1, especially for the 5 kinematic inputs. The most influential factor is now $v_I$ because it has the highest independent and full sensitivity indexes. It accounts for about 28% of the variance of the model outputs by itself, and 58% of the variance in combination with other factors. The most influential factor in the real case, i.e., $\Delta x$, is now the second most influential factor in the hypothetical case. Both the independent and full sensitivity indexes of $\Delta x$ are decreased, for example, $S_T^{\Delta x}$ has dropped from 0.55 to 0.40. The factor $acc_i$ is identified as a non-influential factor in the hypothetical case because its independent and dependent total sensitivity indexes are less than 0.05. Evidently, the SA results in the hypothetical case are incorrect, and this is due to the fact that the correlations of the five kinematic inputs are erroneously modeled. This again serves as a caution that treating dependent factors as independent in SA can yield significant errors in the conclusions. Therefore, to avoid mistakes when performing SA with traffic simulation models, the practitioners should not only consider the marginal distributions of individual parameters, but should also take the global distribution and dependence structure into account.

### 7.4 Summary of the Chapter

This chapter proposes a quantitative method for the SA of models with dependent parameters. It utilizes the copula-based sampling approach introduced in Chapter 6 to generate dependent random samples with known marginal distributions and dependence structure, and estimates the variance-based sensitivity indexes by using the extended Sobol’ formulas (Kucherenko et al., 2012; Mara et al., 2015) and QMC simulation. A case study of the Wiedemann-74 car-following model is included to illustrate the application of the proposed SA method. The results show that the proposed SA can correctly derive the independent and full sensitivity
Figure 7.2: Estimated independent and full sensitivity indexes for the 17 factors of Wiedemann-74 car-following model in the hypothetical case (dependent kinematic inputs are assumed to be independent). In both figures, bars are the mean values, and error bars correspond to the bounds of 90% BCIs of the estimation.

(a) Independent sensitivity indexes for the 17 factors (independent kinematic inputs).

(b) Full sensitivity indexes for the 17 factors (independent kinematic inputs).

indexes for all 17 model factors, in which 5 factors are dependent. The SA results can be used for optimizing model calibration (e.g., through setting priorities of calibration parameters), and for reducing the uncertainties in simulations (e.g., through managing the uncertainties of certain model inputs).

Compared to the classic variance-based SA which is only applicable to models with independent parameters (Söböl’, 1993), the proposed SA approach is more advanced as it can quantify the sensitivity of both independent and dependent parameters. Moreover, the application of the copula-based sampling approach introduced in Chapter 6 makes it possible to link the marginal distributions of individual parameters with the global distribution of all parameters and their dependence structure. Such linkage is vital for SA, as it has been shown in the case study that if the global distribution and dependence structure is ignored or incorrectly modeled (e.g., considering dependent parameters as independent), the SA will yield incorrect results. This also highlights the importance of the proposed SA approach, given the fact that no similar method can be found in the existing literature, or in practices within the transportation research community.

To summarize, the approach proposed in this study applies a state-of-the-art technique for the quantitative SA of both independent and dependent parameters with arbitrary distributions
Chapter 7. Quantitative Sensitivity Analysis for Model with Dependent Parameters

and dependence structure. It is a general global SA approach independent of any specific model. Hence, there will be no difficulty to implement this method as a standard SA tool for traffic simulation models or any complex models in the wider scientific community.

One minor concern in the practical application of this approach is related to its efficiency. Since this approach generally requires more model evaluations than the classic variance-based SA, when the simulation model itself is high dimensional and/or computationally expensive, it might be difficult or even unfeasible to apply this approach directly. A possible solution is to employ a screening approach for reducing the dimension of the model in advance. More details will be given in the next chapter.
Chapter 8

Screening Approach for Model with Dependent Parameters

This chapter is based on the following research paper:

8.1 Introduction

It has been shown in Chapter 7 that only a few recent studies are able to perform SA of models with dependent parameters. For example, studies such as Xu and Gertner (2008), Da Veiga et al. (2009), Li et al. (2010), Xu (2013), and Zhang et al. (2015a) discussed the variance decomposition with parametric or non-parametric methods for models with dependent parameters. Mara and Tarantola (2012), Kucherenko et al. (2012), and Mara et al. (2015) extended the analytical formulations and the corresponding numerical estimators to compute Sobol’s sensitivity indexes (Sobol’, 1993) for dependent parameters. All these methods are quantitative methods, i.e., they quantitatively decompose and analyze the variance contributions of dependent parameters. Yet, for complex models in practice, the quantitative SA might still be difficult or unfeasible to apply when e.g.:  
1) the model is a black-box (e.g., the internal function is not accessible), hence it does not allow any analytical calculation such as the multi-dimensional integrals;  
2) the model contains many parameters and is computationally expensive; or  
3) many parameters have non-standard marginal distributions, and the information regarding their conditional distributions is difficult or even unfeasible to obtain.

For this matter, it is reasonable to recall the use of the qualitative SA, i.e., parameter screening, before performing the quantitative SA. It is shown in Chapters 3 and 4 that for high dimensional and computationally expensive models with independent parameters, the parameter screening can actually enhance the efficiency of the quantitative SA without decreasing the accuracy. Therefore, it is expected that a proper screening method may also provide such benefit for models with dependent parameters. In addition, when the purpose of the SA is not to quantify the effects caused by certain parameters in an exact way, but to identify influential and non-influential parameters or to set priorities for the parameters, it is also reasonable to consider a qualitative screening method rather than a quantitative but computationally more demanding method.

The objective of this chapter is to develop an efficient, non-parametric approach for screening dependent parameters. The performance of the proposed screening method is evaluated through two case studies. The screening results of each experiment are cross-compared with the reference results obtained by the extended variance-based SA introduced in Chapter 7. Results show that the proposed screening approach can efficiently rank the dependent parameters and identify the non-influential parameters, with satisfactory accuracy.

The rest of this chapter is organized as follows. The methodology of the proposed screening method is illustrated in Section 8.2. The details about the case studies, and the corresponding results are introduced and discussed in Section 8.3. Conclusions are presented in Section 8.4.

8.2 Methodology

Sobol’ and Kucherenko (2009), and Kücherenko et al. (2009) suggested that a very high accuracy may not be required in the initial parameter screening, especially when the main purpose is to determine the non-influential parameters or to rank the parameters. Therefore, a qualitative analysis for parameter screening, which considers both efficiency and accuracy,
can be very valuable for the SA practice (as already seen in Chapter 3). A non-parametric method for screening dependent parameters is proposed in this section. This method extends the classic EE method in a similar way as the extended variance-based SA presented in Chapter 7. The details are introduced below.

### 8.2.1 Extended EE for dependent parameters

The variance decomposition for dependent parameters has been introduced in Section 7.2.1, and four sensitivity indexes are provided: independent main sensitivity index $S_{\text{ind}}^i$, independent total sensitivity index $S_{\text{full}}^i$, dependent main sensitivity index $S_{\text{full}}^i$, and dependent total sensitivity index $S_{\text{full}}^i$. The aim of preliminary parameter screening is to identify and remove the non-influential parameters, and to keep the influential or potentially influential parameters for the subsequent SA and/or the model calibration. As neither of the two main sensitivity indexes $S_{\text{ind}}^i$ and $S_{\text{full}}^i$ introduced in Section 7.2.1 includes the interaction effects, they are not appropriate measures for screening non-influential parameters. For instance, small $S_{\text{ind}}^i$ and $S_{\text{full}}^i$ do not necessarily mean that $X_i$ is not influential, as $X_i$ may have higher order interactions with other parameters. Therefore, in the cases with potentially dependent parameters, it is more reasonable to employ the two total sensitivity indexes $S_{\text{full}}^i$ and $S_{\text{full}}^i$ for the initial parameter screening, i.e., a parameter is identified as non-influential when both $S_{\text{full}}^i$ and $S_{\text{full}}^i$ are close to zero.

However, as already discussed in Chapter 7, the computation of the sensitivity indexes is based on the QMC estimation, which usually needs a large number of model runs. Hence, when the model is computationally expensive and/or contains many parameters, it may be unfeasible to perform the QMC estimation directly.

On the other hand, it is proven in Chapters 3 and 4 that the EE method is an accurate and computationally efficient approach for parameter screening. The major problem of the classic EE method is that the effects due to parameters dependence are not considered in the original definition of EE. For example, when a parameter is varied with $\Delta$, the other parameter which is correlated with this parameter should simultaneously have certain variations. However, such variations are not included in Equation (3.1). Therefore, applying the classic EE method directly to models with dependent parameters would yield incorrect screening results.

To overcome the above problem, the classic EE method needs to be extended so that it can cope with models with dependent parameters. Two extended elementary effects are proposed below:

a) independent elementary effects $EE_{\text{ind}}^i$: elementary effects excluding the dependent contribution of parameter $X_i$ (i.e., the impacts due to the dependence between $X_i$ and other parameters); and

b) dependent elementary effects $EE_{\text{full}}^i$: elementary effects including the dependent contribution of parameter $X_i$.

The mathematical definitions of $EE_{\text{ind}}^i$ and $EE_{\text{full}}^i$ follow the similar definitions of $S_{\text{full}}^i$ and $S_{\text{full}}^i$ in Equations (7.4c) and (7.4d), as well as the corresponding QMC estimators in

---

1Recall that the computational cost for $S_{\text{full}}^i$ and $S_{\text{full}}^i$ is $3kN$, where $k$ is the number of parameters, and $N$ is the sample size of the MC experiment (for details see Section 7.2.3).
Equations (7.5d) and (7.5e). For a model \( y = f(X) \), the formulas for computing the extended EE of parameter \( X_i \) are shown below:

\[
EE_{i}^{\text{ind}} = \frac{f(\bar{X}_i', X_{-i}) - f(X_i, X_{-i})}{\Delta}, \tag{8.1a}
\]

\[
EE_{i}^{\text{full}} = \frac{f(X_i', \bar{X}_{-i}) - f(X_i, X_{-i})}{\Delta}. \tag{8.1b}
\]

In the above formulas, \( X_{-i} \) represents all parameters except \( X_i \), i.e., \( X = \{X_i, X_{-i}\} \). \( \{X_i, X_{-i}\} \) is a dependent random vector generated from the joint PDF \( p(X) \). Let \( \{X_i', X_{-i}'\} \) be a different dependent random vector generated from the same joint PDF \( p(X) \). In other words, \( X_i \) and \( X_i' (i \in \{1, k\}) \) are independent of each other, but their marginal PDFs are the same (i.e., \( p(X_i) = p(X_i') \)). \( \{X_i', X_{-i}\} \) is a dependent random vector generated based on a given \( X_{-i} \), i.e., it is generated from the conditional PDF \( p(X_i | X_{-i}) \). \( \{X_i', X_{-i}\} \) is a random vector generated based on a given \( X_i' \), i.e., it is generated from the conditional PDF \( p(X_{-i}' | X_i') \). In addition, according to Bayes’ theorem, \( p(X_i' | X_{-i}) = p(X)/p(X') = p(X)/p(X) = p(X_{-i}|X_i) \).

As already mentioned in Section 3.2.2, in order to explore the global sensitivity, the computation of EE should be performed in the entire input space using multiple (e.g., \( N \)) random samples. The absolute mean \( \mu^* \) and standard deviation \( \sigma \) of the corresponding \( EE_{i}^{\text{ind}} \) and \( EE_{i}^{\text{full}} \) can therefore be used for screening parameters. Specifically, \( X_i \) is identified as:

a) a non-influential parameter when both \( \mu_{i}^{\text{ind}} \approx 0 \) and \( \mu_{i}^{\text{full}} \approx 0 \);

b) an influential parameter when \( \mu_{i}^{\text{ind}} \gg 0 \), i.e., its influence is due to its own contribution (low \( \sigma_{i}^{\text{ind}} \)), or due to the interactions with other parameters (high \( \sigma_{i}^{\text{ind}} \)); or

c) an indirectly influential parameter when \( \mu_{i}^{\text{ind}} \approx 0 \) but \( \mu_{i}^{\text{full}} \gg 0 \), i.e., its influence is mainly due to its dependence with other influential parameters.

The extended EE method comprises two consecutive steps: first, generate independent random samples from uniform distributions; second, transform the independent random samples into unconditional or conditional dependent random samples as required by Equations (8.1a) and (8.1b), and compute the corresponding \( EE_{i}^{\text{ind}} \) and \( EE_{i}^{\text{full}} \). The details about the sampling design and the computation of the extended EEs are given below.

### 8.2.2 Sampling design

As already discussed in Section 3.2.2, different sampling designs (e.g., trajectory design, radial design, and cell design) can be used to overcome the drawback of OAT design, and enhance the computational efficiency of the classic Morris EE method. The case studies in Section 3.4 have shown that the trajectory design (i.e., the quasi-OT design) is an effective sampling design in screening independent parameters. However, as the SA becomes much more complex when the model contains dependent parameters, it is still unclear which sampling design has better performance in practice. Therefore, both the trajectory design and the radial design are employed to generate \( N \) independent random samples. The cell design is not considered here because it is generally less accurate than the other two sampling designs according to the empirical study in [Campolongo et al. (2011)](https://doi.org/10.1007/978-3-642-17756-2_8).

Table 8.1 gives an example of the trajectory design and the radial design for the \( r \)-th random sample (\( r \in [1, N] \)).
which are derived by varying and combining the coordinates from Table 8.1: Example of trajectory design and radial design.

\[ \mathbf{p}_{i,r} = a_{1,r}^i, a_{2,r}^i, a_{3,r}^i, \ldots, a_{k,r}^i \]

\[ \mathbf{b}_{i,r} = b_{1,r}^i, b_{2,r}^i, b_{3,r}^i, \ldots, b_{k,r}^i \]

In Table 8.1, a point represents a \( k \)-dimensional vector, and each coordinate of this point represents a possible value of the corresponding parameter. \( a_r \) (i.e., \( \{a_{1,r}, a_{2,r}, \ldots, a_{k,r}\} \)) and \( b_r \) (i.e., \( \{b_{1,r}, b_{2,r}, \ldots, b_{k,r}\} \)) are two points randomly sampled in a \( k \)-dimensional unit hypercube \( [0, 1]^k \). The random sample from both designs is formed by \( k + 1 \) points, which are derived by varying and combining the coordinates from \( a_r \) and \( b_r \). Specifically, when using the trajectory design, there is exactly one coordinate difference between any two consecutive points on the trajectory\(^2\), i.e., \( \mathbf{p}_{i,r} \) and \( \mathbf{p}_{i+1,r} \) just differ at the coordinate of \( X_i \), \( i \in [1, k] \) (also see the illustration of Figure 3.1 in Section 3.2.2). On the other hand, when using the radial design, there is exactly one coordinate difference between the first point \( \mathbf{p}_{1,r} \) and any other point, i.e., \( \mathbf{p}_{1,r} \) and \( \mathbf{p}_{i,r} \) just differ at the coordinate of \( X_{i-1} \), \( i \in [2, k + 1] \).

The Morris sampling strategy (Morris, 1991) is adopted for the trajectory design in this dissertation. This means that the coordinate difference \( \Delta \) is a constant for all trajectories, i.e., \( b_{i,r} = a_{i,r} + \Delta \) for \( \forall i \in [1, k] \) and \( \forall r \in [1, N] \). To make an efficient exploration of the input space, the quasi-OT design introduced in Chapter 3 is used to generate \( N \) random trajectories that have wide dispersion in the input space.

The radial design is implemented based on Sobol’ sequence (Sobol’, 1976). Following the procedure described in Saltelli et al. (2010), a \( N \)-by-\( 2k \) matrix of Sobol’ quasi-random numbers is produced. The first \( k \) elements of the \( r \)-th row in this matrix are taken to form \( a_r \), and the rest \( k \) elements of the same row are used to form \( b_r \). Unlike the aforementioned Morris strategy, the coordinate difference does not have a fixed value throughout the sampling, i.e., \( \Delta \) is not a constant in the radial design.

### 8.2.3 Computation of the extended EE

Let \( \mathcal{T}(\mathbf{p}_{i,r}, \omega) \) be a transformation function. This function transforms a uniformly distributed independent random vector \( \mathbf{p}_{i,r} \) (i.e., the \( i \)-th point in Table 8.1, \( i \in [1, k + 1] \), \( r \in [1, N] \))

\[ \mathbf{p}_{i,r} = a_{1,r}, a_{2,r}, a_{3,r}, \ldots, a_{k,r} \]

\[ \mathbf{b}_{i,r} = b_{1,r}, b_{2,r}, b_{3,r}, \ldots, b_{k,r} \]
into a dependent random vector under given marginal PDFs and dependence structure. \(\omega\) is the shift parameter, \(\omega \in [0, k]\).

This transformation function contains four sequential transformations, i.e., \(\mathcal{T}_1, \mathcal{T}_2, \mathcal{T}_3,\) and \(\mathcal{T}_4\):

\[
\mathcal{T}(\mathbf{p}_{i,r}, \omega) = \mathcal{T}_4(\mathcal{T}_3(\mathcal{T}_2(\mathcal{T}_1(\mathbf{p}_{i,r}, \omega))))
\]  \hspace{1cm} (8.2)

The specific process of \(\mathcal{T}(\mathbf{p}_{i,r}, \omega)\) is explained below:

1. \(\mathcal{T}_1(\mathbf{p}_{i,r}, \omega)\): shift the first \(\omega\) elements of \(\mathbf{p}_{i,r}\) to its rear, and hence generate a uniformly distributed independent vector \(\mathbf{p}^1_{i,r}\). Specifically, when \(\omega = 0\) or \(k\), \(\mathbf{p}^1_{i,r}\) is the same as \(\mathbf{p}_{i,r}\). For the trajectory design in Table 8.1, \(\mathbf{p}^1_{i,r} = [p_{1,r}, \ldots, p_{k,r}'] = [b_{1,r}, \ldots, b_{k-1,r}, a_{k,r}, \ldots, a_{1,r}]\). If \(\omega = i\), then \(\mathbf{p}^1_{i,r} = \mathcal{T}_1(\mathbf{p}_{i,r}, i) = [p_{i+1,r}, \ldots, p_{k,r}, p_{1,r}, \ldots, p_{i-1,r}, a_{i,r}]\).

2. \(\mathcal{T}_2(\mathbf{p}^1_{i,r})\): apply the inverse standard normal CDF \(\Phi^{-1}(\cdot)\) to each element of \(\mathbf{p}^1_{i,r}\), and produce an independent vector \(\mathbf{p}^2_{i,r}\) that follows \(k\)-dimensional standard normal distribution. Thus, in the above example, \(\mathbf{p}^2_{i,r} = \mathcal{T}_2(\mathbf{p}^1_{i,r}) = [z_{i+1,r}, \ldots, z_{k,r}, z_{1,r}, \ldots, z_{i,r}]\) with \(z_{i,r} = \Phi^{-1}(p_{i,r})\).

3. \(\mathcal{T}_3(\mathbf{p}^2_{i,r})\): apply the IC procedure (see Section 6.2.3) to transform \(\mathbf{p}^2_{i,r}\) into a new vector \(\mathbf{p}^3_{i,r}\) according to the given marginal PDFs and SRCCs. As a result, \(\mathbf{p}^3_{i,r} = \mathcal{T}_3(\mathbf{p}^2_{i,r}) = [x_{i+1,r}, \ldots, x_{k,r}, x_{1,r}, \ldots, x_{i,r}]\).

4. \(\mathcal{T}_4(\mathbf{p}^3_{i,r})\): shift the first \((k - \omega)\) elements of \(\mathbf{p}^3_{i,r}\) back to its rear, so that the elements of the new vector, i.e., \(\mathbf{p}^4_{i,r}\), are in the same order as the elements of \(\mathbf{p}_{i,r}\). In this step, \(\mathbf{p}^4_{i,r} = \mathcal{T}_4(\mathbf{p}^3_{i,r}) = [x_{1,r}, \ldots, x_{k,r}]\).

When the trajectory design is adopted, the required dependent random samples for deriving \(EE^\text{ind}_i\) and \(EE^\text{full}_i\) can be generated based on the points \(\mathbf{p}_{i,r}\) and \(\mathbf{p}_{i+1,r}\). Specifically, the \(r\)-th random samples corresponding to the unconditional dependent vector \{\(X_i, X_{-i}\)\} are generated based on \(\mathbf{p}_{i,r}\), while the \(r\)-th random samples corresponding to the conditional dependent vectors \{\(\bar{X}_{i}', X_{-i}\)\} and \{\(X_i', \bar{X}_{-i}\)\} are generated based on \(\mathbf{p}_{i+1,r}\).

As an example, according to Equation (8.1a), two sets of dependent random samples of \{\(\bar{X}_{i}', X_{-i}\)\} and \{\(X_i, X_{-i}\)\} are needed in the computation of \(EE^\text{ind}_i\). The desired samples of \{\(\bar{X}_{i}', X_{-i}\)\} and \{\(X_i, X_{-i}\)\} should be the same in all dimensions except the \(i\)-th dimension. The transformation function \(\mathcal{T}(\cdot, i)\) is therefore applied on \(\mathbf{p}_{i+1,r}\) and \(\mathbf{p}_{i,r}\) to generate the \(r\)-th desired random samples of \{\(\bar{X}_{i}', X_{-i}\)\} and \{\(X_i, X_{-i}\)\}, i.e., \(\mathcal{T}(\mathbf{p}_{i+1,r}, i)\) and \(\mathcal{T}(\mathbf{p}_{i,r}, i)\), respectively. Consequently, the \(r\)-th independent EE of \(X_i\), i.e., \(EE^\text{ind}_{i,r}\), is computed as:

\[
EE^\text{ind}_{i,r} = \frac{f \left( \mathcal{T}(\mathbf{p}_{i+1,r}, i) \right) - f \left( \mathcal{T}(\mathbf{p}_{i,r}, i) \right)}{\Delta}.
\]  \hspace{1cm} (8.3)

The random samples required for computing \(EE^\text{full}_i\) can be generated similarly using the transformation function \(\mathcal{T}(\cdot, i - 1)\). Accordingly, the \(r\)-th dependent EE of \(X_i\), i.e., \(EE^\text{full}_{i,r}\), can be derived as:
The computation of the independent and dependent EEs can be similarly performed if using the independent random samples generated by the radial design. Let \( q_1, r \) and \( q_{i+1}, r \) denote two vectors that represent the first and \((i + 1)\)-th points in the \( r \)-th radial sample, respectively. The independent and dependent EEs based on the samples under the radial design are derived as:

\[
EE_{i,r}^{\text{ind}} = \frac{f\left(\mathcal{T}(p_{i+1}, i - 1)\right) - f\left(\mathcal{T}(p_i, i - 1)\right)}{b_{i,r} - a_{i,r}},
\]

\[
EE_{i,r}^{\text{full}} = \frac{f\left(\mathcal{T}(q_{i+1}, i)\right) - f\left(\mathcal{T}(q_1, i)\right)}{b_{i,r} - a_{i,r}}.
\] (8.4)

The simulation runs required by the samples from the trajectory design and the radial design are \( N(3k + 1) \) and \( 3k N \), respectively.

### 8.3 Case studies

This section includes two case studies for illustrating the application of the extended EE method. The first case study employs numerical test functions, and the second case study uses the same Wiedemann-74 car-following model presented in Section 7.3. The details are given below.

#### 8.3.1 Case study 1: numerical test functions

In the first case study the extended EE method is tested with three numerical functions. To validate the accuracy of the extended EE method, the extended variance-based SA introduced in Chapter 7 with analytical formulas (i.e., Equations (7.4c) and (7.4d)) and QMC estimators (i.e., Equations (7.5d) and (7.5e)) are used to produce reference results.

**Test functions**

1. **Linear function**

   Consider a simple linear function with three parameters, i.e.,
   \[
   f(X_1, X_2, X_3) = X_1 + X_2 + X_3.
   \]
   This test function has been used in e.g., Kucherenko et al. (2012) and Mara et al. (2015), for testing the performance of the extended variance-based SA approach with different parameters correlations. It is assumed that the parameters have joint normal distribution \( \mathcal{N}(0, \Sigma) \), where

   \[
   \Sigma = \begin{bmatrix}
   1 & \rho_{12} & \rho_{13} \\
   \rho_{12} & 1 & \rho_{23} \\
   \rho_{13} & \rho_{23} & 1
   \end{bmatrix}.
   \]
Moreover, this numerical experiment includes two test scenarios using different sets of correlation coefficients. In the first scenario, positive correlations are considered, i.e., \([\rho_{12}, \rho_{13}, \rho_{23}] = [0.9, 0.4, 0.01]\). In the second scenario, it is assumed that negative correlations exist for parameter pairs \(X_1\)-\(X_2\) and \(X_1\)-\(X_3\), i.e., \([\rho_{12}, \rho_{13}, \rho_{23}] = [-0.9, -0.4, 0.01]\).

(2) Nonlinear function

A nonlinear function with four parameters, i.e., \(f(X_1, X_2, X_3, X_4) = X_1 \cdot X_3 + X_2 \cdot X_4\), is employed in the second numerical experiment. This test function is used in e.g., Kucherenko et al. (2012) for testing the performance of the extended variance-based SA.

For demonstration purposes, this experiment considers a more complex scenario, in which the four parameters are assumed to have four different marginal distributions:

- \(X_1\) has the normal distribution \(N(0, 1)\);
- \(X_2\) has the gamma distribution \(\Gamma(2, 1)\);
- \(X_3\) has the uniform distribution \(U[0, 1]\); and
- \(X_4\) has the log-normal distribution \(\ln N(0, 1)\).

In addition, it is assumed that \(X_1\) and \(X_4\) are correlated with \(\rho_{14} = 0.8\), and \(X_3\) and \(X_4\) are correlated with \(\rho_{34} = 0.3\). The correlation coefficients for other parameter pairs are arbitrarily set to 0.

(3) G function

The third numerical experiment employs the G function (see the introduction in Section 5.3.1) with 12 parameters: \(G = \prod_{i=1}^{12} \frac{|4X_i - 2| + a_i}{1 + a_i}\).

All parameters have uniform distributions, i.e., \(X_i \sim U[0, 1], i \in [1, 12]\). For the ease of demonstration, the same setting for the coefficients \(a_i\) as those provided in Campolongo et al. (2007) is employed: \([a_1, \ldots, a_{12}] = [0.001, 89.9, 5.54, 42.1, 0.78, 1.26, 0.04, 0.79, 74.51, 4.32, 82.51, 41.62]\).

It is assumed that \(X_1\) and \(X_2\) have positive correlation \(\rho_{12} = 0.8\), and \(X_7\) and \(X_9\) have negative correlation \(\rho_{79} = -0.8\). The correlation coefficients for other parameter pairs are arbitrarily set to 0.

8.3.1.2 Experimental design

For each test function, the extended EE method is applied first to derive \(\mu^*\) and \(\sigma\) of \(EE^{ind}\) and \(EE^{full}\). Both trajectory design and radial design are employed in the sampling process. Then the results are cross-compared with \(S^{T, ind}\) and \(S^{T, full}\) computed by the extended variance-based SA. The details are provided below.

(1) Linear function

This experiment arbitrarily employs 90 random samples generated by the trajectory design (i.e., \(N = 90\)), and 100 random samples generated by the radial design (i.e., \(N = 100\)). Hence, the two designs have the same number of model runs, i.e., 900 runs.
The analytical values of $S^{T,\text{ind}}$ and $S^{T,\text{full}}$ for each parameter are computed using Equations (7.4c) and (7.4d).

(2) Nonlinear function

The experiment of the nonlinear function uses 240 random samples generated by the trajectory design (i.e., $N = 240$), and 260 random samples generated by the radial design (i.e., $N = 260$). In total, both designs run the model 3,120 times.

Since the analytical computation of $S^{T,\text{ind}}$ and $S^{T,\text{full}}$ is rather complicated when the four parameters are from different marginal distributions, the QMC estimators given by Equations (7.5d) and (7.5e) are used instead. The estimation utilizes a size of 10,000 Sobol’ quasi-random numbers, hence the total computational cost of the QMC estimation is $3kN = 120,000$.

(3) G function

The numerical experiment of the G function uses 390 random samples generated by the trajectory design (i.e., $N = 390$), and 400 random samples generated by the radial design (i.e., $N = 400$). Hence, the total computational cost required by these two sampling designs are 14,430 and 14,400, respectively.

Due to the complexity of the G function and the high dimensionality, the sensitivity indexes are difficult to compute using analytical formulas in this experiment. Thus, the QMC estimator is also used here to produce the reference results. Again a size of 10,000 Sobol’ quasi-random numbers is used for the estimation, which results in a total computational cost of $3kN = 360,000$ model runs.

The computational cost (i.e., number of model runs) required by the extended EE method and the extended variance-based SA for each test function are reported in Table 8.2.

Table 8.2: Number of model runs required by the extended EE method and the extended variance-based SA.

<table>
<thead>
<tr>
<th>Function</th>
<th>Number of extended</th>
<th>Number of extended</th>
<th>Variance-based SA</th>
</tr>
</thead>
<tbody>
<tr>
<td>$X_1 + X_2 + X_3$</td>
<td>3</td>
<td>900 (TD, RD)</td>
<td>–</td>
</tr>
<tr>
<td>$X_1 \cdot X_3 + X_3 \cdot X_4$</td>
<td>4</td>
<td>3,120 (TD, RD)</td>
<td>120,000</td>
</tr>
<tr>
<td>G function</td>
<td>12</td>
<td>14,430 (TD, 14,400 (RD)</td>
<td>360,000</td>
</tr>
</tbody>
</table>

Note: TD = trajectory design, RD = radial design.

8.3.1.3 Results of case study 1

(1) Linear function

The results corresponding to the two test scenarios are presented in Figures 8.1 and 8.2, respectively.
Figure 8.1 shows that the extended EE method based on the trajectory design and the radial design yield the same parameter ranking in the first scenario. Specifically, if only considering the independent variance contribution (see Figures 8.1(a) and 8.1(c)), $X_3$ is the most influential parameter, and $X_1$ is the least influential parameter. On the other hand, when considering the full variance contributions including the dependent contribution (Figures 8.1(b) and 8.1(d)), the extended EE method always identifies $X_1$ as the most influential parameter, and $X_3$ as the least influential parameter. The above finding is consistent with the ranks based on the analytical values of $S^T, ind$ and $S^T, full$ in Figure 8.1(e): $S^T, ind_1 < S^T, ind_2 < S^T, ind_3$ and $S^T, full_3 < S^T, full_2 < S^T, full_1$. These results highlight the importance of using both independent and dependent elementary effects for screening parameters, as the judgment on influential or non-influential parameters can bias if only one measure is used. In this scenario, as none of the three parameters has both $(\mu^ind, \sigma^ind)$ and $(\mu^full, \sigma^full)$ close to 0, they are all selected as influential parameters and should not be eliminated by the initial parameter screening.

Figure 8.1: Results of numerical experiment 1: $f(X_1, X_2, X_3) = X_1 + X_2 + X_3$, $\{X_1, X_2, X_3\} \sim N(0, 1), [\rho_{12}, \rho_{13}, \rho_{23}] = [0.9, 0.4, 0.01]$.

(a) Plots of $\mu^i$ versus $\sigma$ of the independent EE (trajectory design).

(b) Plots of $\mu^i$ versus $\sigma$ of the dependent EE (trajectory design).

(c) Plots of $\mu^i$ versus $\sigma$ of the independent EE (radial design).

(d) Plots of $\mu^i$ versus $\sigma$ of the dependent EE (radial design).

(e) Analytical sensitivity indexes.

The second test scenario adopts a different set of correlation coefficients. Figure 8.2 shows that the analysis based on the trajectory design again gives the same parameter ranking as the analysis based on the radial design. In addition, the parameter rankings obtained by both designs are also consistent with the ranking obtained by the extended variance-based approach. In this scenario, parameter $X_3$ is the most influential parameter, with or without
Figure 8.2: Results of numerical experiment 1: \( f(X_1, X_2, X_3) = X_1 + X_2 + X_3, \{X_1, X_2, X_3\} \sim \mathcal{N}(0, 1), [\rho_{12}, \rho_{13}, \rho_{23}] = [-0.9, -0.4, 0.01] \).

(a) Plots of \( \mu^* \) versus \( \sigma \) of the independent EE (trajectory design).

(b) Plots of \( \mu^* \) versus \( \sigma \) of the dependent EE (trajectory design).

(c) Plots of \( \mu^* \) versus \( \sigma \) of the independent EE (radial design).

(d) Plots of \( \mu^* \) versus \( \sigma \) of the dependent EE (radial design).

(e) Analytical sensitivity indexes.

considering its dependent contribution. Parameter \( X_1 \) has low \( \mu^{*,\text{ind}} \) but relatively high \( \mu^{*,\text{full}} \), which means that its full variance contribution including the dependent contribution is not negligible. Therefore, to avoid making Type II error in parameter screening, \( X_1 \) is still considered as an influential parameter. The least influential parameter is \( X_2 \) in this scenario, as both \( (\mu^{*,\text{ind}}, \sigma^{\text{ind}}) \) and \( (\mu^{*,\text{full}}, \sigma^{\text{full}}) \) are very small. Consequently, \( X_2 \) can be fixed without influencing the results in the subsequent analysis.

It is worth mentioning that in the first scenario, \( EE_{i}^{\text{ind}} \) is always smaller than \( EE_{i}^{\text{full}} \) for \( \forall i \in [1, 3] \). It means that under this dependence structure, the impact due to parameter dependence always has a positive contribution to the total variance. On the other hand, in the second scenario, the impact of parameter dependence is negative for \( X_2 \), which results in \( EE_{2}^{\text{ind}} > EE_{2}^{\text{full}} \).

The results presented above indicate that the parameter ranking determined by the extended EE method match well with the ranking computed via analytical formulas. Therefore, this experiment proves the accuracy of the extended EE method in ranking the dependent parameters, while only a small number of model runs are required by both sampling designs.
(2) Nonlinear function

The results of the second numerical experiment are illustrated in Figure 8.3.

Figure 8.3: Results of numerical experiment 2: $f(X_1, X_2, X_3, X_4) = X_1 \cdot X_3 + X_2 \cdot X_4$. $X_1 \sim N(0, 1)$, $X_2 \sim \Gamma(2, 1)$, $X_3 \sim \mathcal{U}[0, 1]$, $X_4 \sim \ln N(0, 1)$, $\rho_{14} = 0.8$, $\rho_{34} = 0.3$.

(a) Plots of $\mu^*$ versus $\sigma$ of the independent EE (trajectory design).

(b) Plots of $\mu^*$ versus $\sigma$ of the dependent EE (trajectory design).

(c) Plots of $\mu^*$ versus $\sigma$ of the independent EE (radial design).

(d) Plots of $\mu^*$ versus $\sigma$ of the dependent EE (radial design).

(e) Estimated sensitivity indexes.

Screening results in Figures 8.3(a) and 8.3(c) show that if not considering the dependent contribution, parameter $X_2$ is the most influential parameter, and parameters $X_1$ and $X_3$ are the least influential parameters. If the dependent contribution is considered (Figures 8.3(b) and 8.3(d)), parameter $X_4$ is the most influential parameter, and parameter $X_3$ is the least influential parameter. The screening results hence suggest that under the given dependence structure, $X_3$ is the least influential parameter in all cases, and it can be eliminated in the subsequent analysis. Moreover, although $X_1$ is one of the least influential parameters in Figures 8.3(a) and 8.3(c), since it has a strong correlation with the most influential parameter $X_4$, its dependent contribution should not be ignored. In other words, $X_1$ is also considered as an influential parameter under the given dependence structure.

In general, the findings from the screening analysis in this experiment are consistent with the results of the QMC estimation in Figure 8.3(e). Notice that there is a minor difference in parameter ranking when using the two sampling designs. When considering the dependent contribution, $X_2$ seems to be more influential than $X_1$ if the trajectory design is used for
sampling (Figure 8.3(b)), while the analysis based on the radial design suggests that $X_1$ is more influential than $X_2$ (Figure 8.3(d)). According to the reference results obtained by the QMC estimation, it is clear that the ranking obtained by the radial design is correct. This may be explained by the fact that for any parameter, the radial design does not require a fixed variation (i.e., $\Delta$ in Equations (8.1a) and (8.1b)) of the corresponding random samples, which could result in a better exploration of the input space than the trajectory design. Such feature is more desirable for the case when both the input space and the model are nonlinear, and hence the extended EE method with the radial design is expected to yield more accurate results. Nonetheless, since the ranking results obtained by the trajectory design are valid for the most and least influential parameters, and they do not contain any Type I or Type II error, the trajectory design can still be adopted in initial parameter screening.

(3) **G function**

The results of the experiment of the G function are presented in Figure 8.4. It is found that independent of the sampling design used, $X_1$, $X_5$, $X_6$, $X_7$, and $X_8$ always have high $(\mu^{\text{ind}}, \sigma^{\text{ind}})$ and $(\mu^{\text{full}}, \sigma^{\text{full}})$. Hence, they are the influential parameters, with or without considering the dependent contribution. In Figures 8.4(a) and 8.4(c), $X_2$ and $X_9$ have the lowest $(\mu^{\text{ind}}, \sigma^{\text{ind}})$, thus they are the least important parameters if the dependent contribution is not considered. However, these two parameters have rather high $(\mu^{\text{full}, \sigma^{\text{full}}})$ in Figures 8.4(b) and 8.4(d). It indicates that the dependent contribution of these two parameters are not trivial. As a result, they are also identified as influential parameters in this case. $X_4$, $X_{11}$, and $X_{12}$ always have very low $(\mu^{\text{ind}}, \sigma^{\text{ind}})$ and $(\mu^{\text{full}, \sigma^{\text{full}}})$, therefore they are selected as non-influential parameters, and they can be fixed without influencing the SA results in the subsequent analysis.

The above findings can be explained by the fact that in the G function, a parameter with a small coefficient $a_i$ will be an influential parameter, and vice versa. In this case, the coefficients $a_4$, $a_5$, $a_6$, $a_7$, and $a_8$ are much smaller than the other coefficients, thus it is certain that the corresponding parameters $X_1$, $X_5$, $X_6$, $X_7$, and $X_8$ are influential parameters. On the contrary, parameters $X_4$, $X_{11}$, and $X_{12}$ are independent of other influential parameters, and the corresponding coefficients $a_4$, $a_{11}$, and $a_{12}$ are rather high, thus they should be selected as non-influential parameters. The coefficients $a_2$ and $a_9$ are also high, which suggests that the corresponding parameters $X_2$ and $X_9$ would be the non-influential parameters if their dependent contributions are not considered. However, in the given dependence structure, $X_2$ and $X_9$ have strong correlations with the two most influential parameters $X_1$ and $X_7$, respectively. As a result, the variations of $X_2$ and $X_9$ will also bring variations to $X_1$ and $X_7$, i.e., $X_2$ and $X_9$ have indirect variance contributions to the model output. Therefore, $X_2$ and $X_9$ are also identified as influential parameters in this case.

When taking the sensitivity indexes computed by the QMC estimation (Figure 8.4(e)) as references, and cross comparing the results obtained by the two sampling designs, it is again found that the results obtained by the trajectory design are a little less accurate than the results obtained by the radial design. For instance, $X_7$ is less influential than $X_5$ and $X_8$ in Figure 8.4(a), while Figure 8.4(c) shows $X_7$ is more influential than the other two parameters. According to the ranking by $S^2$ (Figure 8.4(e)), the results obtained by the radial design in Figure 8.4(c) is correct. Nonetheless, if considering that the inconsistency of the ranking does not produce any Type I or Type II error in selecting the influential and non-influential
Figure 8.4: Results of numerical experiment 3:  \[ G = \prod_{j=1}^{12} \frac{|4X_j - 2| + a_j}{1 + a_j}, \{X_1, \cdots, X_{12}\} \sim \mathcal{U}[0, 1], [a_1, \cdots, a_{12}] = [0.001, 89.9, 5.54, 42.1, 0.78, 1.26, 0.04, 0.79, 74.51, 4.32, 82.51, 41.62], \rho_{12} = 0.8, \rho_{79} = -0.8.

(a) Plots of \( \mu^* \) versus \( \sigma \) of the independent EE (trajectory design).

(b) Plots of \( \mu^* \) versus \( \sigma \) of the dependent EE (trajectory design).

(c) Plots of \( \mu^* \) versus \( \sigma \) of the independent EE (radial design).

(d) Plots of \( \mu^* \) versus \( \sigma \) of the dependent EE (radial design).

(e) Estimated sensitivity indexes.

<table>
<thead>
<tr>
<th>( X_1 )</th>
<th>( X_2 )</th>
<th>( X_3 )</th>
<th>( X_4 )</th>
<th>( X_5 )</th>
<th>( X_6 )</th>
<th>( X_7 )</th>
<th>( X_8 )</th>
<th>( X_9 )</th>
<th>( X_{10} )</th>
<th>( X_{11} )</th>
<th>( X_{12} )</th>
</tr>
</thead>
<tbody>
<tr>
<td>( S_{T, ind} )</td>
<td>( S_{T, full} )</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

parameters, the screening results obtained by the trajectory design can still be regarded as satisfactory.

### 8.3.1.4 Summary of case study 1

To sum up, the three numerical experiments show that the extended EE method has satisfactory accuracy in screening the parameters of a model that contains both dependent and independent parameters. In particular, the screening results, i.e., the sets of influential and non-influential parameters, obtained by this method are consistent with the reference results obtained by the extended variance-based SA. In addition, the extended EE method with the radial design always yields the same parameter ranking as the reference results for all test cases. The extended EE with the trajectory design also performs well in all tests, although there are some errors in parameter ranking in the experiments of the nonlinear function and the G function. However, as in these two experiments the errors of parameter ranking are not relevant for the selection of the most and least influential parameters (i.e., there is no Type I or Type II error), such errors could be considered trivial for the preliminary screening analysis.

Moreover, if compared with the extended variance-based SA, which is in general more
accurate but also requires a large number of model runs (e.g., 120,000 for the experiments 2 and 3, respectively), the extended EE method has a comparable accuracy in parameter screening, but it requires much less computational cost (e.g., 3,120 and 14,400 runs for numerical experiments 2 and 3, respectively). Therefore, because of its high efficiency in computation, this method is especially useful for the initial parameter screening of high dimensional and/or computationally expensive models which contain both independent and dependent parameters.

Last but not least, the proposed method is distribution free, and it can work even without the knowledge of the internal mathematical function of the model, or the joint/conditional distributions of the parameters. This feature is very attractive for the SA of many commercial simulation models, which often behave like black-boxes and/or contain parameters from different arbitrary distributions.

8.3.2 Case study 2: sequential SA with dependent parameters

The numerical experiments in the first case study have proven the accuracy and efficiency of the extended EE method. The second case study performs a sequential SA\(^3\) for the Wiedemann-74 car-following model. Specifically, the extended EE method with both trajectory and radial designs is used to identify the least influential parameters, then the extended variance-based SA introduced in Chapter 7 is performed to quantify the variance contributions of the influential parameters. The final results are cross-compared with the SA results found in Section 7.3. The objective is to check whether the extended EE method can aid the extended variance-based SA in terms of improving the computation efficiency but without reducing the accuracy.

8.3.2.1 Experimental design

For the ease of cross-comparing SA results, the same SA factors as those analyzed in the case study in Section 7.3 are included here. In total there are 17 factors, including 5 dependent kinematic inputs (i.e., \(\Delta x\), \(v_f\), \(v_l\), \(acc_f\), and \(acc_l\)) and 12 independent model parameters (i.e., \(BX_1\), \(BX_2\), \(EX_1\), \(EX_2\), \(CX_0\), \(BMAX_1\), \(BMIN_1\), \(BNULL_1\), \(VDES\), \(R_1\), \(R_2\), and \(NR\)). The distributions and correlation coefficients of these factors are already reported in Figure 6.1, Table 6.1, and Table 7.1, respectively.

8.3.2.2 Results of case study 2

(1) Results of the extended EE method

The initial screening of the 17 factors is based on 250 random samples generated from the trajectory design (i.e., \(N = 250\)), and 256 random samples generated from the radial design (i.e., \(N = 256\)). In total, the initial screening based on the trajectory design requires 13,000 model runs, while the initial screening based on the radial design requires 13,056 model runs.

\(^3\) Note that the sequential SA performed in this case study is able to deal with dependent parameters, hence it is different with the sequential SA (i.e., quasi-OTEE plus Kriging-based SA) introduced in Section 5.3.2, which only deals with independent parameters. Accordingly, the factors analyzed in this case study are not the same as those analyzed in the case study in Section 5.3.2, in which the dependent kinematic inputs are grouped as one independent factor in the SA.
Chapter 8. Screening Approach for Model with Dependent Parameters

Figure 8.5: Screening results for the 17 factors based on the trajectory design.

(a) Plots of $\mu^*$ versus $\sigma$ of the independent EE.

(b) Plots of $\mu^*$ versus $\sigma$ of the dependent EE.

(c) Plots of $\mu$ versus $\sigma$ of the independent EE.

(d) Plots of $\mu$ versus $\sigma$ of the dependent EE.

Note: Lines in Figures 8.5(c) and 8.5(d) correspond to $\mu = \pm 2SEM$.
Figure 8.6: Screening results for the 17 factors based on the radial design.

(a) Plots of $\mu^*$ versus $\sigma$ of the independent EE.

(b) Plots of $\mu^*$ versus $\sigma$ of the dependent EE.

(c) Plots of $\mu$ versus $\sigma$ of the independent EE.

(d) Plots of $\mu$ versus $\sigma$ of the dependent EE.

Note: lines in Figures 8.6(c) and 8.6(d) correspond to $\mu = \pm 2$SEM.
Figures 8.5 and 8.6 present the results of the extended EE method with the trajectory design and the radial design, respectively. Figures 8.5(a), 8.5(b), 8.6(a) and 8.6(b) show that with both sampling designs, the four dependent factors, i.e., $\Delta x$, $v_f$, $v_l$, and $acc_f$, all have high $(\mu^{\text{ind}}, \sigma^{\text{ind}})$ and $(\mu^{\text{full}}, \sigma^{\text{full}})$. Therefore, they are identified as influential factors, with or without considering their dependent contributions. The factor $acc_l$ has very low $(\mu^{\text{ind}}, \sigma^{\text{ind}})$ but high $(\mu^{\text{full}}, \sigma^{\text{full}})$ using both sampling designs. It means that if not considering the dependent contribution, $acc_l$ is certainly a non-influential factor. However, since $acc_l$ is correlated with other influential factors (mostly with $acc_f$), its dependent contribution is not negligible. In other words, due to the correlation, fixing the variance of $acc_l$ will indirectly reduce the variances of other influential factors, which may lead to wrong results in the subsequent SA or model calibration. Therefore, $acc_l$ should also be included as an influential factor.

Figures 8.5 and 8.6 also show that the most influential independent factor is $BM1N_1$, which always have higher $\mu^{\text{ind}}$ and $\mu^{\text{full}}$ than the other independent factors. On the other hand, factors $BNUL1_1$, $VDES$, $BMAX_1$, and $EX_2$ always have the lowest $(\mu^{\text{ind}}, \sigma^{\text{ind}})$ and $(\mu^{\text{full}}, \sigma^{\text{full}})$ among all factors, therefore these factors are identified as non-influential factors. Note that these four factors are independent factors, hence they do not have any dependent contributions to the variance of model output. Accordingly, they can be fixed in the subsequent SA.

Figures 8.5(c), 8.5(d), 8.6(c) and 8.6(d) show that with both the trajectory design and the radial designs, the scatter plots of factors $BX_1$, $BX_2$ and $EX_1$ are always outside the wedge formed by $\mu = \pm 2\text{SEM}$. Therefore, these factors should be considered as influential factors. The independent factors $CX_0$, $NR$, $R_1$, and $R_2$ fall inside the wedge. However, the high $\sigma^{\text{ind}}$ and $\sigma^{\text{full}}$ suggest that they may have strong non-linear interaction effects with other factors, therefore, they are also selected as influential factors.

In addition, when making a further comparison between $\mu^{\text{ind}}-\mu^{\text{ind}}$, and between $\mu^{\text{full}}-\mu^{\text{full}}$ for all factors in both Figures 8.5 and 8.6, it is found that the independent factor $BM1N_1$ has negative $\mu^{\text{ind}}$ and $\mu^{\text{full}}$, but $|\mu^{\text{ind}}| = \mu^{\text{ind}}$ and $|\mu^{\text{full}}| = \mu^{\text{full}}$. This suggests that $BM1N_1$ always have negative $EE^{\text{ind}}$ and $EE^{\text{full}}$. In other words, increasing the value of $BM1N_1$ will always result in the decrease of the predicted acceleration rate, and vice versa. A similar inference can be made for $\Delta x$, for which most EEs are positive, i.e., it is highly possible that increasing the value of $\Delta x$ will simultaneously increase the predicted acceleration rate. This finding makes perfect sense for the car-following process, as the increase of the gap between two vehicles will very likely make the follower increase his acceleration rate in the subsequent time step.

Overall, 4 factors, i.e., $BNUL1_1$, $VDES$, $BMAX_1$, and $EX_2$, are selected as non-influential factors, and they will be fixed to their mean values in the extended variance-based SA. This finding is consistent with the SA results presented in Section 7.3.2. Moreover, if using the independent and dependent total sensitivity indexes $T^{\text{ind}}$ and $T^{\text{full}}$ from Figure 7.1 as references for factor ranking, it is again found that the analysis based on the radial design is more accurate than the analysis based on the trajectory design, especially for the ranks of the most influential factors. For instance, when considering the dependent contribution, the ranks of $\Delta x$ and $v_f$ are different in Figures 8.5(b) and 8.6(b). According to Figure 7.1, $\Delta x$ is less influential than $v_f$ in this case, and hence the rank obtained by the radial design is correct. However, considering that both sampling designs are able to yield correct screening
results, and the computational cost is less than 20% of the computational cost required by the extended variance-based SA, they are both recommended in the experimental design for initial parameter screening.

(2) Results of the extended variance-based SA

Overall, 13 factors, i.e., \(\Delta x, v_f, v_l, acc_f, acc_l, BX_1, BX_2, EX_1, CX_0, BMIN_1, R_1, R_2,\) and \(NR\), are analyzed by the extended variance-based SA. The independent and full sensitivity indexes are estimated 1,000 times via bootstrapping. It is found that with 512 quasi-random samples (i.e., \(N = 512\)), the SA results are sufficiently stable (i.e., the BCIs are narrow enough). Thus the computational cost of the extended variance-based SA is \(4kN = 26,624\).

Figure 8.7 presents the results of the extended variance-based SA for the 13 factors. It shows \(\Delta x\) is the most influential factor, which accounts alone for about 20% of the variance of the model output, and 52% of the total variance if including the effects due to its interactions and dependence with other factors.

Figure 8.7: Estimated independent and full sensitivity indexes for the 13 factors of Wiedemann-74 car-following model. In both figures, bars are the mean values, and error bars correspond to the bounds of 90% BCIs of the estimation.

The second most influential factor is \(v_l\), which accounts for almost 56% of the total variance if considering the main effect, the interaction effects, and the effects due to the dependence with other parameters. The 3rd to 13th influential factors are \(v_f, BMIN_1, acc_f, acc_l\), \(EX_1, BX_1, CX_0, BX_2, NR, R_1\) and \(R_2\). Because the \(ST, ind\) and \(ST, full\) of all factors are above 0.05, none of these 13 factors should be identified as a non-influential factor.

A cross-comparison between Figures 7.1 and 8.7 shows that the sensitivity indexes of the 13 influential factors are almost the same as those obtained in Section 7.3.2. In other
words, reducing the model dimension through eliminating the non-influential factors from the beginning does not introduce any significant changes to the sensitivity indexes of the influential factors. On the other hand, the initial parameter screening helps the sensitivity indexes getting converge faster with less random samples (e.g., this case only requires 512 random samples to achieve stable sensitivity indexes, while the case study in Section 7.3 requires 1,024 random samples).

Finally, Table 8.3 gives a summary of all case studies that have employed the Wiedemann-74 car-following model in this dissertation (Chapters 5, 7 and 8). It shows that all methods are able to identify the most influential factors, i.e., the kinematic inputs and the model parameter $BM1N_1$, which in total account for over 90% of variance of the model outputs. Therefore, all SA results obtained in these case studies are valid.

Among the four SA methods, the classic variance-based SA is the least efficient approach, and it is not able to provide the sensitivity information for dependent factors. The sequential SA (i.e., the quasi-OTEE plus the Kriging-based SA for independent factors) requires the least computational cost. This is mainly due to a combined application of parameter grouping, parameter screening, and the metamodel. However, same as the classic variance-based SA, this approach also fails to detect the sensitivity information for individual kinematic inputs.

The extended variance-based SA and the extended EE method are more powerful as they are able to deal with the dependence contribution from individual parameters. Yet, these two approaches generally require more computational cost than the aforementioned sequential SA. On the other hand, when cross-comparing the total computational cost spent in the case studies in Chapter 7 and in this chapter, it is found that 45% of the total computational cost can be saved by adopting the parameter screening, while the sensitivity indexes of the influential factors are almost the same.

The above finding again highlights the benefit of implementing initial parameter screening for the SA of high dimensional and/or computationally expensive models. Therefore, an efficient strategy for the SA of high dimensional and computationally expensive models with both independent and dependent parameters could be:

1. grouping the dependent parameters as one independent parameter;
2. using the quasi-OTEE to eliminate the non-influential independent parameters;
3. using the extended EE method to further screen the remaining independent and dependent parameters; and
4. using the extended variance-based SA to derive the sensitivity indexes based on the simulation model or the metamodel.

More information regarding how to perform an efficient SA in practice is provided in Chapter 9.

### 8.4 Summary of the Chapter

This chapter introduces a non-parametric approach that extends the classic EE method for screening dependent parameters. Based on the extended sensitivity indexes for dependent parameters, which were initially introduced in Kucherenko et al. (2012) and Mara et al. (2015), two extended elementary effects are proposed: the independent elementary effects $EE_{ind}$ and the dependent elementary effects $EE_{full}^{\text{full}}$. To compute the sensitivity measures (i.e., $\mu$, $\mu^*$, and
Table 8.3: Summary of the reviewed SA approaches.

<table>
<thead>
<tr>
<th></th>
<th>Chapter 5</th>
<th>Chapter 7</th>
<th>Chapter 8</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Classic Variance-based SA</td>
<td>Quasi-OTEE + Kriging-based SA</td>
<td>Extended Variance-based SA</td>
</tr>
<tr>
<td>Number of independent factors</td>
<td>$25^{(a)}$</td>
<td>$25^{(a)}$</td>
<td>12</td>
</tr>
<tr>
<td>Number of dependent factors</td>
<td>N/A</td>
<td>N/A</td>
<td>5</td>
</tr>
<tr>
<td>Total computational cost for factor screening</td>
<td>N/A</td>
<td>2,600</td>
<td>N/A</td>
</tr>
<tr>
<td>Total computational cost for variance quantification</td>
<td>135,000</td>
<td>1,024$^{(b)}$</td>
<td>69,632</td>
</tr>
<tr>
<td>Total computational cost</td>
<td>135,000</td>
<td>3,624</td>
<td>69,632</td>
</tr>
<tr>
<td>5 most influential factors</td>
<td>$Kin, BMIN_1, EX_1, NR, R_2$</td>
<td>$Kin, BMIN_1, NR, BX_1, EX_1$</td>
<td>$\Delta x, v_f, BMIN_1, acc_f$</td>
</tr>
</tbody>
</table>

TD: trajectory design; RD: radial design.

$^{(a)}$The independent factors include 1 group factor $Kin$ ($Kin = \{x_f, x_l, v_f, v_l, acc_f, acc_l, L_1\}$).

$^{(b)}$It is the cost for deriving the Kriging metamodel.
Chapter 8. Screening Approach for Model with Dependent Parameters

of $EE^{ind}$ and $EE^{full}$), two sampling designs, i.e., the trajectory design and the radial design, are implemented together with the IC procedure (see Chapter 6) in order to generate dependent random samples. A qualitative analysis of the corresponding sensitivity measures can thus be performed to distinguish the corresponding parameter as a non-influential parameter when both $\mu^{*ind}$ and $\mu^{*full}$ are close to 0, or an influential parameter in all other cases. Moreover, when $\mu^{*ind} \approx 0$ but $\mu^{*full} \gg 0$, the corresponding parameter can be considered as indirectly influential due to its dependence with other influential parameters.

The performance of the extended EE method is empirically evaluated via two case studies, one based on numerical experiments, and the other based on the Wiedemann-74 car-following model. In the numerical experiments, the screening results are cross-compared with the extended sensitivity indexes $S^{T,ind}$ and $S^{T,full}$, which are computed either analytically or through the QMC estimations. All test results have shown that the extended EE method can properly distinguish the influential and non-influential parameters, even when the parameters have different marginal distributions (e.g., the experiment of the nonlinear function), or when the model contains many dependent and independent parameters (e.g., the experiment of the G function). The parameter rankings are generally consistent with the results obtained through the extended variance-based SA, yet the screening approach requires much less model runs. For instance, in the experiment of the G function, the extended EE method requires about 25 times less model runs than the QMC estimator.

The second case study sequentially applies the extended EE method and the variance-based approach introduced in Chapter 7 for the SA of the Wiedemann-74 model. The screening results as well as the final sensitivity indexes of the influential factors are consistent with the results presented in Section 7.3.2, while over 45% of the total computational cost is saved through the sequential analysis.

To sum up, because of its distribution free feature and efficiency in computation, the extended EE method is very attractive for the purpose of preliminary screening of dependent parameters, especially when the model is high dimensional and/or computationally expensive. In these cases, a direct application of the extended variance-based SA will be rather time consuming or even unfeasible. Hence, it is recommended to perform the qualitative parameter screening beforehand in order to reduce the number of relevant parameters, and conduct the quantitative variance-based SA sequentially based on the screening results.
Chapter 9

Application Framework for Global Sensitivity Analysis

9.1 Introduction

In this dissertation both qualitative and quantitative approaches are developed in order to conduct efficient SA for complex simulation models. The performance of these approaches is already proven by multiple case studies. The findings so far have demonstrated that global SA is a powerful tool for e.g., identifying non-influential parameters, quantifying the variance contributions from multiple parameters, analyzing the influence from parameter dependence. Therefore, global SA can be considered as an essential step in optimizing the model calibration, especially when the model is computationally expensive and/or contains many parameters.

However, most global SA techniques introduced above are not so straightforward to be implemented in practice. Moreover, there are usually many technical issues which need to be solved before starting the SA (e.g., the computer code to run the model automatically). As a result, performing a proper SA with traffic simulation models is still considered as a challenging task in practice (Pünzo and Ciuffo, 2014b). It is worth remembering that many users of the traffic simulation model are neither full-time researchers nor experts with in-depth experience in SA and calibration problems. It is not difficult to imagine that without proper guidance for the application of global SA, those inexperienced users might be prone to skipping the SA, or choosing a simple but problematic approach (such as the OAT-based approach) in their applications. In the end, it could lead to multiple issues in model calibration, including but not limited to, incomplete set of calibration parameters, unrealistic values, waste of computational resources.

This chapter aims to provide a general framework for supporting the application of global SA methods in the calibration of traffic simulation models. The objective is to provide pragmatic information for practitioners, especially for those who do not have much experience or knowledge with advanced SA techniques, to ease the difficulties in e.g., selecting the proper SA method, setting up the SA procedure, analyzing the SA outcomes. More details are given in the following section.
Chapter 9. Application Framework for Global Sensitivity Analysis

9.2 Application Framework

The application framework is developed based on the knowledge gained from both theoretical and practical studies conducted within this dissertation, as well as the information provided in relevant literature such as Antoniou et al. (2014), Daamen et al. (2014), and Saltelli et al. (2008). This framework includes 7 steps: (1) defining the objective of the SA; (2) identifying relevant parameters; (3) selecting the SA method; (4) designing the sampling process; (5) evaluating the model; (6) computing the SA results; and (7) checking the validity of the SA results. A schematic illustration of the whole framework is presented in Figure 9.1. The details of each step are explained below.

9.2.1 Defining the objective of the SA

As already discussed in Chapter 2, different SA approaches employ different statistic measures for deriving the sensitivity information, hence it is very likely that different approaches will not give exactly the same SA results even for the same simulation model. Therefore, before choosing any specific method for performing the SA, the practitioner should have a clear definition of the objective. Such objectives may vary on a case by case basis, and they are directly related to the specific requirements of the model calibration.

For instance, when the model contains more than e.g., 30 parameters, a reasonable objective of the SA can be set to distinguish the non-influential parameters, and reduce the size of potential parameters for calibration. The reason is most traffic simulation models are over-parameterized, and only a small set of key parameters are actually responsible for the total variations of the model output (Antoniou et al., 2014). Thus, calibrating the full set of parameters can be unnecessary or even unfeasible, especially when there are constraints on the whole project budget (e.g., time, manpower). Accordingly, in this case the SA target is to qualitatively screen the non-influential parameters, and fix these parameters to any nominal values within the corresponding boundaries. In other words, this SA objective corresponds to the Factor Fixing setting (see Section 2.2.8).

On the other hand, when the total number of calibration parameters is small (e.g., less than 10), the SA objective can be set to analyze the variance contributions of the influential parameters, and rank the influential parameters based on the quantitative sensitivity information. This is because if the most influential parameters are not properly calibrated and fixed from the beginning, no matter how well the other parameters are calibrated, a small variation of the influential parameters could totally change the model outputs, and thus make the calibrated simulation results meaningless. By ranking the parameters based on their variance contributions, the practitioner can set priorities in the calibration, i.e., first calibrating the most influential parameter, then the second most influential parameter, and so on for the other ones. The above SA objective therefore leads to the Factor Prioritization setting (see Section 2.2.8).

Last but not least, when the aim of the calibration is to make the simulation results fall within or beyond given thresholds (e.g., make the simulated travel time on specific links fall below certain values), the corresponding objective of the SA can be set to identify the invalid regions in the input space of the corresponding parameters. In this way, the practitioner can avoid spending precious computational resources in exploring those invalid regions, and thus...
9.2. Application Framework

Figure 9.1: Schematic illustration of the application framework for global SA in calibrating traffic simulation models.

START

Define the Objective

- Qualitative analysis
  - Settings: FF, FP, FM

Select SA Method

- Quasi-OTEE
- Kriging-based SA
- Variance-based SA

Identify Relevant Parameters

<table>
<thead>
<tr>
<th>Dependence</th>
<th>Distribution</th>
</tr>
</thead>
<tbody>
<tr>
<td>1 0.5 0</td>
<td>+ 0.5 0.8</td>
</tr>
<tr>
<td>0 0.8 1</td>
<td>+ 0.8 1</td>
</tr>
</tbody>
</table>

Evaluate the Model

Model Outputs

- Random samples for relevant parameters
- Other fixed parameters

Model

Model Inputs

f(x_1, x_2, ..., x_N)
f(x_1, x_2, ..., x_N)
f(x_1, x_2, ..., x_N)

Design Sampling Process

Independent samples
Dependent samples
Sample size

Check the Validity

- Are the SA results meaningful?
  - Yes
  - No

- Are the SA results stable?
  - Yes
  - No

Compute SA Results

- Variance
- Standard deviation
- Confidence interval

END
optimize the calibration process. Consequently, this SA objective is linked to the Factor Mapping setting (see Section 2.2.8).

Finally, it is worth mentioning that the objective of the SA may also influence the decision about which parameters are included in the analysis. For example, if the objective of the SA is to determine whether the traffic demands (i.e., OD pairs) or the driving behavior parameters (including parameters from both car-following and lane-changing models) are more influential on e.g., the total delay in a network, the SA can be performed based on two groups of parameters: one group includes all OD pairs, and the other includes all driving behavior parameters. On the contrary, if the objective of the SA is to identify which car-following parameters are less influential than the other car-following parameters with respect to e.g., the travel time on links, the SA can then be performed for all individual car-following parameters.

9.2.2 Identifying relevant parameters

The discussion from previous chapters have shown that the dimension of the model (i.e., the number of parameters) is a key factor that directly influences the computational cost of the SA. Therefore, to make the SA of a high dimensional model affordable, it is necessary to identify the relevant parameters at a very early stage.

As an example, in the case study presented in Section 3.4.3, the total number of parameters contained in the Vissim model is 192. In general, no SA technique can be applied directly to analyze such high number of parameters. Therefore, to make the SA feasible, an initial screening is performed to identify the relevant parameters. The judgment is made based on e.g., the features of the simulation model, empirical data, relevant research, commonsense, as well as existing experience. In the end, the number of relevant parameters in this case study is reduced to 14, and the SA is successfully performed.

However, the above parameter screening is not based on any analytical or numerical computation, indeed, it is a rather subjective process. It is possible that the arbitrary judgment may actually include non-influential parameters (i.e., Type I error) and/or eliminate influential parameters (i.e., Type II error). The possibility of such risk might be high when the parameters have high-order interactions and/or dependence. Therefore, the practitioner should not obsessively pursue the minimized set of relevant parameters in this step. As a rule-of-thumb, a reasonable size for the relevant parameters can be around 20 to 50 if the computational cost per model run is less than 1 hour.

Furthermore, it is also very important to define the proper distributions (including the upper and lower boundaries) and dependence structure for the relevant parameters in this step. In practice, the choice of distribution is usually based on literature, relevant research, empirical data, expert opinions, or even reasonable assumptions (Punzo and Ciuffo, 2014a). In absence of a priori information on the marginal distribution of a parameter, it is often assumed that this parameter has a uniform distribution, and the boundaries are arbitrarily set based on the parameter’s default value (e.g., ±50% of the default value). However, the assumption of uniform distribution should be cautiously used because the type of the distribution function could have significant impacts on the SA results. For example, the formulas for computing the sensitivity indexes in Equations (7.4a) to (7.4d) are dependent on the marginal and joint PDFs, hence different distribution functions will certainly yield different sensitivity indexes. Additionally, an extremely narrow range of variation is not recommended because it actually
restricts the variability of the corresponding parameter, and turns the global SA into a local analysis. On the contrary, an excessively wide range of variation is also not necessary since it requires much more computational efforts.

The importance of correctly modeling the dependence structure in SA has already been explained in Chapter 7. As in the case of the marginal distributions, the dependence structure (e.g., the correlation matrix) can also be obtained from relevant studies, field measurements, or reasonable assumptions when no a priori information is available. Then the sensitivity of the dependent parameters can be analyzed using the approaches introduced in Chapters 7 and 8. One tricky issue regarding the parameter dependence is that in some cases, some “independent” parameters might be actually dependent when they are sampled from a truncated input space (especially when MCF is used). For instance, a vehicle’s maximum acceleration $acc_{\text{max}}$ should always be larger than its minimum acceleration $acc_{\text{min}}$. Therefore, a valid sample of $acc_{\text{max}}$ should always be generated from the region where it is not less than $acc_{\text{min}}$, i.e., the value of $acc_{\text{max}}$ is dependent on the value of $acc_{\text{min}}$. One possible solution for modeling this dependence structure, as suggested by Saltelli et al. (2008), is to approximate such dependencies as “explicit relationships with a noise term”. In the above example, one can introduce an independent non-negative noise term $\Delta$. Then $acc_{\text{max}}$ is represented as $acc_{\text{min}} + \Delta$ in the SA, and the approaches for independent parameters can be used. However, it should be noticed that the SA will only be able to provide the sensitivity information for $acc_{\text{min}}$ and $\Delta$. In order to know the sensitivity of $acc_{\text{max}}$, the practitioner needs to conduct further analysis to aggregate the sensitivity information of $acc_{\text{min}}$ and $\Delta$.

Last but not least, as all decisions made in this step are mainly based on the practitioner’s own choice, to make the analysis more reliable, it is always recommended to check the validity of the results when the SA is finished. If the results are not satisfactory, the practitioner might need to adjust the set of relevant parameters, the distributions, as well as the dependence structure according to the available SA results, and start a new SA (see Section 9.2.7).

9.2.3 Selecting the SA method

So far several global SA methods have been reviewed (see Chapter 2) and developed (see Chapters 3 to 8) in this dissertation. It has been shown that different SA methods have different features, and there is no method that universally performs better than the other methods for all SA problems. In other words, the choice of the most appropriate method depends upon the specific SA objective and the characteristics of the model (e.g., dimension, computational cost).

Some general recommendations with respect to the selection of the SA method are given below:

a) If the SA objective is to qualitatively identify the non-influential parameters (i.e., FF setting) from a complex model (e.g., black-box), the practitioner may consider the quasi-OTEE approach if there is no dependent parameter, or the extended EE approach if the model contains dependent parameters. The computational cost is $N(k + 1)$ for the quasi-OTEE, $N(3k + 1)$ for the extended EE with trajectory design, and $3kN$ for the extended EE with radial design (recall that $k$ is the model dimension, and $N$ may vary from 10 to few hundreds).

For cases when the model contains many parameters (e.g., over 50) and the computational
cost per simulation is high (e.g., over 1 hour), one feasible approach is to group the parameters based on their similarities (Ciuffo and Azevedo, 2014), and conduct the parameter screening for groups. The group operation is also recommended for computationally expensive models with dependent parameters due to the fact that the extended EE approach requires much more model runs than the quasi-OTEE. In this case, the dependent parameters can be grouped as one independent parameter. The quasi-OTEE is first applied to eliminate the non-influential independent parameters, and then the extended EE is used to screen the remaining parameters (including both independent and dependent parameters).

b) If the SA objective is to analyze the variance contribution, and to identify the most influential parameters (FP setting) from a complex model, the practitioner may consider the classic variance-based SA for models with only independent parameters, or the extended variance-based SA for models with both independent and dependent parameters. The corresponding computational cost is $N(k + 2)$ for the classic variance-based SA, and $4kN$ for the extended variance-based SA (again $k$ is the model dimension, and $N$ could vary from $10^3$ to $10^5$).

For cases when the model has a high dimension, the direct application of the variance-based SA would result in a huge number of model runs. In such case, the practitioner may conduct a sequential analysis, i.e., first use the aforementioned screening approach to reduce the model dimension, and then perform the variance-based SA with the simplified model. Another feasible option, which can be applied together with the screening approach, is to develop the metamodel and conduct the variance-based SA with the metamodel. Since the metamodel normally only needs a rather short time to run, the computational cost is mainly the time spent in developing the metamodel itself (e.g., from $2^5$ to $2^{12}$ runs for developing the Kriging model using Sobol’ sequence).

Moreover, as the extended variance-based SA requires way more computational cost than the classic variance-based SA, when the model itself is computationally expensive, it is again more feasible to group the dependent parameters together as an independent parameter, and conduct the sequential SA dedicated to independent parameters (i.e., a combination of the quasi-OTEE and the Kriging-based SA, see the case study in Section 5.3.2). When the model dimension is further reduced to e.g., less than 10, the sequential approach for dependent parameters (i.e., extended EE plus the extended variance-based SA, see the case study in Section 8.3.2) can be adopted.

c) If the SA objective is to identify the critical region in which the variations of certain parameters will make the simulation results fall beyond certain thresholds (i.e., FM setting), the MCF approach introduced in Chapter 2 can be used. According to Saltelli et al. (2008), the computational cost may vary from $500$ to $2,000$ runs (see Table 2.1).

d) If the model is linear or quasi-linear, the derivative-based SA (e.g., SND) and regression-based SA introduced in Chapter 2 are recommended due to their low computational demand. If the model only contains a small number of independent parameters (e.g., less than 10), and the practitioner only cares about the qualitative sensitivity information such as the parameter ranking, the scatter plot is recommended because despite its simplicity, it allows an intuitive presentation of the sensitivity information.
9.2.4 Designing the sampling process

The sampling process is required to generate random samples for computing the relevant sensitivity measures (e.g., elementary effects, Sobol’ sensitivity indexes). The decision about which sampling approach to implement is related to the SA method chosen in the previous step. In this dissertation, two types of sampling design are discussed: the sampling design for independent parameters, and the sampling design for dependent parameters.

Often, different parameters in a complex simulation model can have different types of distributions, and/or different ranges under the same distribution. To simplify the sampling design for independent parameters, it is a common practice to first generate independent random samples in a $k$-dimensional unit hypercube $\mathbb{H}^k = [0, 1]^k$, and later use the isoprobabilistic transformations (Lemaire et al., 2010) to transform the random samples from the uniform distribution $\mathcal{U}[0, 1]$ to the original distribution of the corresponding parameter. The trajectory design (see Sections 3.3 and 8.2.2), the radial design (see Section 8.2.2), the sampling design based on quasi-random sequence (see Section 2.2.5), as well as other practical sampling designs such as LHS (Saltelli et al., 2009), can be adopted for the purpose of generating the random samples in $\mathbb{H}^k$.

It is worth noting that when a parameter has a distribution with infinite support (i.e., distributions for which the tails go to infinity), a reasonable design could consider cutting the tails of the distribution and generating samples from the truncated support. For instance, suppose a parameter has a normal distribution $\mathcal{N}(\mu = 0.5, \sigma = 0.15)$, the practitioner may cut the distribution at the points 0.11 and 0.89, which correspond to the 0.5th and 99.5th percentiles of $\mathcal{N}(0.5, 0.15)$, respectively. In this way, the truncated support $[0.11, 0.89]$ contains 99% of the possible values from the original distribution $\mathcal{N}(0.5, 0.15)$.

The sampling design for dependent parameters is extensively discussed in Chapter 6. In general, the practitioner can start by generating independent random samples in $\mathbb{H}^k$ using the aforementioned approaches, then apply the IC procedure to transform the independent random samples to the dependent samples under given distributions and dependence structure. The design based on Sobol’ quasi-random sequence for sampling unconditional and conditional dependent samples, which is required for the extended variance-based SA for dependent parameters, is introduced in Section 7.2.3.

In addition to the sampling approach, the practitioner should also select a reasonable sample size in this step. This is, however, rather tricky since a small sample size may lead to a poor exploration of the input space, while a large sample size requires significant computational cost. Since there is no universal rule that defines the optimum sample size, a feasible way in practice is to start with a small sample size in the SA, and check later the stability of the SA results. If necessary, increase the sample size and continue the analysis (more details will be discussed in Section 9.2.7).

9.2.5 Evaluating the model

In this step the model will be evaluated iteratively to compute the corresponding model outputs required by the SA.

The inputs of the model include the random samples obtained for all relevant parameters, as well as those non-relevant parameters, which have fixed values in the evaluation. In
practice, the practitioner may need to develop a computer program to automatically load the inputs, change the values of the corresponding parameters, and run the model continuously. This is because without the aid from a proper computer program, the practitioner might have to run the model manually over e.g., thousands of times.

Developing such computer program might not be very sophisticated for a mathematical model (such as the Wiedemann-74 model), or an open source traffic package (e.g., SUMO, see DLR, 2016). However, many commercial packages such as Vissim and Aimsun only provide limited APIs (Application Program Interface) to their users, and there are usually many restrictions on the redevelopment of the model. As a result, not all parameters in these commercial packages are allowed to be modified by the user directly. For instance, the APIs of the lane-changing parameters (i.e., Parameters 4 to 12 in Table 3.3) are not open in Vissim 5.40, and thus their values can not be modified during the simulation. This problem can make the whole model evaluation rather challenging for most practitioners (Punzo and Ciuffo, 2014b). In order to make the model evaluation and the subsequent SA possible, the practitioner may need to develop additional codes with some extra efforts. In the example above (i.e., the case study in Section 3.4.3), a computer program was developed in C# for automatically evaluating the Vissim model. Specifically, the values of those lane-changing parameters had to be changed outside Vissim, i.e., through manipulating the Vissim input script. The source code for this program is provided in Chapter A.

Furthermore, many traffic simulators, especially those commercial simulation packages, are able to produce several outputs (e.g., travel time on sections, vehicle counts at detectors) simultaneously with one model evaluation. The decision about which model output(s) to be included in the SA should be made in accordance with the chosen Measure of Performance (MoP) for model calibration. The MoPs are variables describing the status of a system (Law and Kelton, 2000). For the calibration and validation of a traffic simulation model, different MoPs can be adopted depending on the complexity of the system and the features of the specific analysis. Some of the commonly used MoPs are time series of speeds and counts, headways, queue length and turning flows at intersections, network travel time, and vehicle trajectories (more information about the MoPs for calibration is provided in Daamen et al., 2014, pp. 95–97). Accordingly, the model outputs used for the SA can be one single output (e.g., flow, speed, acceleration) or multiple outputs simultaneously.

### 9.2.6 Computing the SA results

The SA results (e.g., EE, Sobol’ sensitivity index, SND) can be computed using the approaches introduced in previous chapters. In case multiple model outputs are produced, the practitioner may need to compute the sensitivity measures for each model output, or for a combination of multiple outputs (i.e., the outputs are aggregated via a predefined function).

In addition, in order to assess the reliability of the obtained SA results, it is recommended to compute the confidence intervals of the corresponding sensitivity measures. The confidence intervals can be estimated using the bootstrapping technique (Efron and Tibshirani, 1993; Archer et al., 1997). The main algorithm of the bootstrapping is explained below.

Suppose a size of 1,024 random samples are generated in order to compute $S_T$, i.e., the total sensitivity index $S_T$ for an independent parameter $X_1$. Accordingly, a set of 1,024 model outputs are produced, i.e., $Y = \{y_1, y_2, \cdots, y_{1024}\}$. A bootstrapping sample set, i.e., $B_1$, can
be generated by re-sampling (i.e., sampling with replacement) from the set \( Y \). Specifically, \( B_1 \) has the same sample size (i.e., 1,024) as \( Y \). Then the bootstrapping estimation of \( S^T_1 \) can be computed based on \( B_1 \), and it is denoted as \( S^T_{1,B_1} \).

The above bootstrapping process can be repeated as many times as the practitioner wants. Suppose 1,000 bootstrapping sample sets are generated, i.e., \( \{B_b\}_{b=1}^{1000} \). Accordingly, based on \( \{B_b\}_{b=1}^{1000} \), 1,000 estimations of \( S^T_1 \) can be produced, i.e., \( \{S^T_{1,B_b}\}_{b=1}^{1000} \). Then the confidence interval based on bootstrapping (i.e., BCI) can be obtained by e.g., taking the specific percentiles from the distribution of \( \{S^T_{1,B_b}\}_{b=1}^{1000} \). For instance, the 90% BCIs shown in Figure 4.2(b) are computed by taking the 5% and 95% percentiles of the corresponding sensitivity indexes as the lower and upper endpoints, respectively.

The practitioner would also need to develop computer programs to conduct the bootstrapping, and compute the SA results automatically. If compared with the computation time spent in evaluating the model, the total time spent in bootstrapping and computing the SA results is expected to be negligible since no further model evaluation is needed in this step. On the other hand, the practitioner might need to spend much time in interpreting the results and making the analysis.

### 9.2.7 Checking the validity of the SA results

The last step in this application framework is to check the validity of the SA results. This step is rather vital as it guarantees the quality of the whole SA.

The validity check includes two parts. The first part is to check if the SA results are stable. To this end, the practitioner could make an empirical analysis of the aforementioned BCIs for the corresponding sensitivity measures. A wide BCI means that there are a lot of uncertainties in the sensitivity measure, i.e., the values of the corresponding sensitivity measure can have many fluctuations depending on the random sample used. However, currently there is no scientific rule that defines at which level the width of the BCI will lead to a stable result. A rule-of-thumb suggested by Zhang et al. (2015b) indicates the SA results can be considered as stable when the BCIs of the most sensitive parameters are less than 10% of the corresponding sensitivity indexes.

In case unstable SA results are obtained, the practitioner is recommended to e.g., increase the sample size, adopt other sampling designs, and then perform a new analysis until the results are stable. In addition, if the new SA only requires to increase the sample size, the outputs from existing model evaluations might be re-used in order to save the computational cost. For instance, to compute the Sobol’ sensitivity indexes, a practitioner might start with 512 simulations based on the first 512 quasi-random numbers in the Sobol’ sequence. If it is found that the sample size is not big enough to achieve stable results, he might try to compute the sensitivity indexes based on e.g., 1,024 random samples. Since the first 512 quasi-random numbers are already used in the simulation, this time he just needs to generate the 513th to 1,024th quasi-random number in the Sobol’ sequence, and performs the model evaluation accordingly. The computation of the new sensitivity indexes will then based on a combination of the 512 existing outputs and the 512 newly generated outputs.

Once the SA results are stable, the second part of the validity check is to check if the results are meaningful and making sense for the corresponding traffic model. If the answer is yes, then the SA can be ended. Otherwise, the practitioner might need to try a different SA
method (e.g., switch from a qualitative method to a more accurate and quantitative method), or even include more parameters as the relevant parameters. Punzo and Ciuffo (2014b, p. 50) provide an example to explain the necessity of this process:

“...one could be interested in understanding the influence of the lane changing parameters on the lane counts, and being surprised by the fact that none of them has a sensible influence on the output chosen. It could be the case that such parameters have influence in their interactions with the car-following parameters that means one needs to include the latter in the analysis, or verify whether they have influence on other output, such as lane speeds.” (Punzo and Ciuffo; 2014b; p. 50)

As the set of relevant parameters and/or the SA method is changed, the practitioner will need to repeat the whole SA from the very beginning, until the SA results are acceptable.

### 9.3 Summary of the Chapter

This chapter proposes a general and pragmatic framework to support the use of global SA in the calibration of traffic simulation models. This framework contains 7 working steps that cover multiple important issues in practice. The goal is to make the global SA approaches accessible to practitioners in order to increase the efficiency and effectiveness of the calibration and validation process.

It is expected that the practitioners, especially those who do no have much experience in SA and calibration, can benefit from this application framework in terms of reducing the difficulties for implementing global SA techniques in the context of traffic simulation models. In addition, the proper usage of the SA methods will not only provide correct parameter sets for model calibration and validation, but also greatly reduce the efforts required for computing and analyzing the results. This should, in turn, increase the number of practitioners who calibrate and validate their models prior to their use.
Chapter 10

Conclusions and Outlook

10.1 Summary

This doctoral research is intended to provide standard and pragmatic SA algorithms for the calibration of microscopic traffic simulation models. Efforts are therefore made throughout this dissertation in order to solve multiple theoretical and practical issues regarding the development and implementation of global SA techniques in the context of traffic simulation models, especially those that are high dimensional and computationally expensive. The major findings and contributions from this dissertation are summarized below.

1. In-depth understanding of global SA methods and existing applications in traffic simulation models

The first contribution of this dissertation is an in-depth understanding of the global SA methods and their applications. A thorough review of the most widely used global SA approaches and existing SA applications in traffic simulations is provided in Chapter 2: Global SA approaches are able to explore the entire input space of model parameters, and hence they allow to detect the interaction effects among multiple parameters. Thus, the global SA approaches are more advanced and reliable than local SA approaches (e.g., the OAT-based approach), which are, however, commonly adopted by practitioners in the transportation research community.

In addition, an extensive study is performed to investigate and understand the principles, application conditions, as well as the the possible outcomes of the following global SA approaches: derivative-based approach (e.g., SND), sampling-based approach (e.g., scatter plot), regression-based approach, screening approach, variance-based approach, metamodel-based approach, and Monte Carlo filtering. A summary of the results is presented in Section 2.2.8: It shows that different statistic measures are employed by different approaches for performing the SA. Therefore, the practitioners should choose the proper SA approach tailored to the specific SA settings (i.e., FP, FF, and FM) and the requirements of the applications.

Finally, the review of existing SA applications in traffic simulations shows that although practitioners are getting more aware of the importance of SA nowadays, the global SA is still not adopted as a common practice in the calibration and validation of traffic models, and several mistakes can be found in the available applications (e.g., adopting a local SA method,
treated dependent parameters as independent). This is mainly because many of the traffic simulation models are rather complex (e.g., high dimensional, black-box, computationally expensive), and there is a shortage of effective SA approaches, especially for models containing dependent parameters. To solve these issues, efficient and accurate SA algorithms, which are able to deal with both the qualitative and quantitative SA of complex simulation models, are required.

2. Development of efficient global SA methods for models with independent parameters

Although multiple global SA methods are extensively introduced in many existing literature (e.g., Saltelli et al., 2008), the efficiency of these methods is rarely discussed. A low-efficiency method might actually be unpractical for the SA of traffic simulation models, given the fact that most simulation models, especially those commercial software packages such as Vissim and Aimsun, usually contain a large amount of parameters and are computationally expensive. To this end, this dissertation contributes to the development of efficient SA methods, including both qualitative and quantitative methods, for models with independent parameters.

- The quasi-OTEE approach for screening non-influential parameters from a high dimensional model is proposed in Chapter 3. This approach is developed based on the classic EE method with significant improvement in terms of the efficiency in sampling. Through computing and qualitatively comparing the $\mu$, $\sigma$ and $\mu^*$ of the EEs, this approach can efficiently identify the influential and non-influential parameters in a complex model with independent parameters.

  The efficiency and accuracy of this approach have been evaluated through three cases studies, including both numerical experiments and real simulation models (i.e., Vissim and Aimsun). It is proven that the quasi-OTEE approach is ideal for the preliminary SA of computationally expensive models, especially when those quantitative, yet more complex SA techniques (e.g., variance-based approaches) are unfeasible. It can be easily implemented as a stand-alone tool when the purpose of the study is to distinguish influential and non-influential parameters. It could also be used to set the foundation for other quantitative techniques if a more refined analysis is desired afterwards (e.g., investigate the non-negligible interactions, identify the correct sensitivity ranks of influential parameters).

- An exploratory study that compares the quasi-OTEE approach and the Kriging-based approach, which was recently developed and applied by Ciuffo et al. (2013) for the quantitative SA, is performed in Chapter 4. The comparison is based on the results from two case studies in Aimsun and Vissim. It is found that both approaches are able to identify, to a good degree, the influential and non-influential parameters. The Kriging-based approach is able to define with very high precision the ranking of the most influential parameters, but at the cost of missing some high dimensional interactions. On the other hand, the quasi-OTEE approach is very robust on all types of interactions, and therefore it is better in identifying those non-influential parameters to discard.

  These findings suggest a joint application of these two approaches for the SA of complex simulation models: the quasi-OTEE SA can be used in the first stage for screening the parameters and deciding which parameters to discard, the Kriging-based SA can be used
in the second stage to refine the analysis and derive the correct rank of the influential parameters.

- A sequential SA approach is implemented by combining the quasi-OTEE approach and the Kriging-based approach in Chapter 5. The sequential SA is able to produce more quantitative results than a stand-alone screening method, while it requires less computational cost in estimating the Kriging metamodel thanks to the initial parameter screening.

The performance of the sequential SA is empirically demonstrated via two case studies including both numerical test functions and the Wiedemann-74 car-following model. The results in all numerical tests show that the sequential SA can properly identify the most influential parameters and their ranks. In addition, the results obtained in the case study of Wiedemann-74 model show that the sequential SA is able to correctly derive the ranks of the most influential factors from the 25 factors of the model, while the number of model evaluations is 40 times lower than that needed by the standard variance-based SA. This feature makes this sequential approach very attractive, especially when dealing with the SA of independent parameters in a high dimensional and computationally expensive model.

### 3. Development of efficient global SA methods for models with dependent parameters

Most SA methods and guidelines for traffic simulation models found in the literature are designed to work with models containing independent parameters only. However, given the fact that there are often many dependent parameters contained in a traffic simulation model, efficient and accurate SA methods dedicated to complex models with dependent parameters are essentially important. For this reason, this dissertation also contributes to the development of both qualitative and quantitative SA methods for models with dependent parameters.

- A general copula-based approach for generating random samples of dependent parameters is proposed in Chapter 6. Specifically, the Gaussian copula and the Spearman’s Rank Correlation Coefficient (SRCC) are employed in the sampling process. This has made this approach attractive in terms of sampling from arbitrary marginal distributions and dependence structure.

A case study is included to demonstrate the copula-based sampling approach by making dependent samples of the kinematic inputs of the Wiedemann-74 car-following model. Results show that the marginal distributions and correlation coefficients of the random samples are comparable with those obtained from empirical car-following data. Thus, this approach can be utilized in the SA of dependent parameters for generating dependent random samples required by e.g., the computation of extended Sobol’ sensitivity indexes and the extended EEs.

- A quantitative SA approach for models with dependent parameters is introduced in Chapter 7. By generating dependent random samples from given arbitrary marginal distributions and dependence structure, this approach is able to estimate the variance-based sensitivity indexes with the extended Sobol’ formulas (Kücherenko et al., 2012; Mará et al., 2015) and the QMC simulation.
The performance of this approach is evaluated through a case study using the Wiedemann-74 model. Results show that this approach can correctly derive the independent and full sensitivity indexes for all 17 model factors, including 5 dependent factors. Moreover, the sensitivity indexes for the 8 influential independent factors are consistent with those obtained by the aforementioned sequential SA.

- A non-parametric screening approach for dependent parameters is developed in Chapter 8. This approach extends the classic EE method and proposes two extended EEs, i.e., the independent elementary effects $EE^{ind}$ and the dependent elementary effects $EE^{full}$.

The performance of this approach is evaluated via two case studies including both numerical experiments and the Wiedemann-74 model. The results of numerical experiments show that this approach can properly distinguish the influential and non-influential parameters, and the parameters ranks are generally consistent with the results obtained through analytical computation or QMC estimation, although it requires e.g., 25 times less model runs than the later. In the case study of the Wiedemann-74 model, the screening results as well as the final sensitivity indexes of the influential factors are consistent with the results presented in Chapter 7, while over 45% of the total computational cost is saved through the sequential application of the extended EE approach and the extended variance-based approach introduced in Chapter 7.

Hence, due to its distribution free feature and efficiency in computation, the extended EE approach is very attractive for the purpose of preliminary parameter screening, especially when the model is computationally expensive and contains several independent and dependent parameters.

It is worth mentioning that all SA approaches developed in this dissertation are general approaches, i.e., they are independent of any specific traffic simulation model. Thus, it is possible to apply them as practical and efficient tools for any computationally expensive model in other disciplines from the wider scientific community, where simulations are also commonly used.

4. Development of a general framework for the application of global SA methods in calibration of traffic simulation models

Finally, this dissertation proposes a general framework for supporting the application of global SA in the calibration of traffic simulation models, and provides pragmatic guidance for implementing the global SA techniques in practice. The application framework is introduced in Chapter 9. It contains seven working steps: (1) defining the objective of the SA; (2) identifying relevant parameters; (3) selecting the SA method; (4) designing the sampling process; (5) evaluating the model; (6) computing the SA results; and (7) checking the validity of the SA results.

Specifically, the following rule-of-thumbs are recommended when implementing the global SA for high dimensional and computationally expensive simulation models:

a) Adopt the Factor Fixing setting for the initial SA, and the Factor Prioritization setting for the subsequent SA;

b) Define the proper distributions and dependence structure for relevant parameters in advance;
10.2 Recommendations for Future Research

This dissertation has developed efficient algorithms, including both qualitative and quantitative approaches, for the SA of complex simulation models. The performance of these algorithms has been proven by the corresponding case studies. Below are some recommendations for possible research extensions.

(i) In this dissertation, the SA algorithms are developed based on the assumption that all parameters have continuous distribution functions over finite supports. However, such assumption is not necessarily true for traffic simulation models, as there are also parameters with discrete distribution functions (e.g., the traffic counts can only have integer values). In addition, because of some constrains, the model with uniformly distributed parameters might be defined in a non-rectangular domain (e.g., a triangular domain for the model \( f(X_1, X_2) \) with \( X_1 \sim U[0,1] \), \( X_2 \sim U[0,1] \), \( X_1 \leq X_2 \)). Applying the SA algorithms provided in this dissertation directly to these models could yield fallacious results. Thus, further efforts are required in, e.g., improving the sampling process, adjusting the formulations of the sensitivity indexes, or even developing new SA algorithms for parameters with special distributions.

(ii) In Chapters 6 to 8 the Gaussian copula is employed for modeling the given dependence structure. There are many other types of copulas, for example, Clayton, Gumbel, and
Frank copulas (Nelsen, 1999). In a calibration study the practitioner is usually interested in the normal values of the dependent parameters, hence the Gaussian copula might be enough in most cases for fitting the dependence structure. However, in other studies the practitioner may also want to investigate the influence of extreme values of certain parameters. Therefore, the research can be devoted to employ other copulas such as the extreme-value copulas (Gudendorf and Segers, 2010) in the sampling process and compute the corresponding sensitivity indexes.

(iii) The SA algorithms developed in this dissertation are independent of any specific traffic simulation model, i.e., they do not rely on any specific theory from transportation research. On the one hand, this means that these algorithms are general mathematical approaches, and they can be applied to other models in the wider scientific community without any further modification. On the other hand, involving available knowledge in the analysis could actually enhance the overall efficiency. For instance, a priori information might show that some parameters are more influential than others. Since the ranking of the least influential parameters is often not important for the calibration, to achieve more accurate SA results without increasing the total computational cost, it is reasonable to spend most model runs in computing the sensitivity indexes of the parameters which are known to be influential. To this end, it would be interesting to explore which available information is valuable for the SA, and develop SA approaches that integrate such information.

(iv) Some recent studies such as Osorio et al. (2015) have implemented a Simulation-based Optimization (SO) technique in the calibration of traffic models. With the aid from metamodels, the simulation-based calibration can be efficiently performed without any advanced sensitivity information. However, it is not clear whether the SA is still able to offer extra benefits when the SO is involved in the calibration. Therefore, future research could be devoted to properly evaluate the value of SA in the presence of SO in model calibration, or to propose a new approach that combines SA and SO for a more efficient calibration.


Ge, Q. and M. Menendez (2012b) Sensitivity analysis for calibrating Vissim in modeling the Zurich network, paper presented at the 12th Swiss Transport Research Conference, Ascona, May 2012.


Ge, Q. and M. Menendez (2016a) Exploring the variance contributions of correlated model parameters: A sampling-based approach and its application in traffic simulation models, manuscript submitted to *Physica A: Statistical Mechanics and its Applications*.


Appendix A

Program for Automatic Simulation in Vissim

A.1 Introduction

This program is used to run the Vissim simulator automatically, so that a Monte Carlo simulation can be performed based on a large number of random samples. The inputs of this program are the trajectories generated from the quasi-OT sampling. The program loads the data (i.e., *.CSV file) of every sampling point from the quasi-OT first, and changes the value of corresponding parameters via the COM interface or via modifying the Vissim input file (i.e., *.inp file) directly. Then it runs the simulations automatically with different parameters value sets. Finally it saves the results (e.g., travel time, average speed, etc.) from each simulation to separate data files (i.e., *.CSV file).

This program is coded in C#, and it should be run in a PC with 64-bit Windows 7 and Vissim 5.40. In addition, the pre-installation of Microsoft .NET Framework 3.5 is required for running the program. The main code is provided below.

A.2 Source Code

```csharp
using System;
using System.Collections.Generic;
using System.Linq;
using System.Windows.Forms;
using System.IO;
using VISSIM_COMSERVERLib;
using System.Data;
using Microsoft.Office.Interop.Excel;
using System.Globalization;

namespace VISSIMSA
{
    static class Program
    {
        /// <summary>
        /// The main entry point for the application.
        /// </summary>
        [STAThread]
        static void Main()
        {
            System.Windows.Forms.Application.SetCompatibleTextRenderingDefault(false);
        }
    }
}
```
public static void VISSIM_RUN(string VISSIMFilePath_str, string VISSIMLayoutPath_str, string CSVDataPath_str)
{
    System.Data.DataTable dt = ReadCSVtoDataTable(CSVDataPath_str);


    string[] inputfile = File.ReadAllLines(VISSIMFilePath_str);
    int length = VISSIMFilePath_str.Length;
    string traveltimepath = VISSIMFilePath_str.Remove(length - 4) + "_traveltime.csv";
    int DrivingBHNumber = 0;
    //this loop is used to find the first line which contains the driving behavior in the VISSIM input file
    while (inputfile[DrivingBHNumber] != "-- Driving Behavior: --")
    {
        DrivingBHNumber++;
    }
    if (File.Exists(traveltimepath))
    {
        File.Delete(traveltimepath);
    }
    int simulationtimes = 1;
    IVissim vis = new VissimClass();
    foreach (DataRow dr in dt.Rows)
    {
        if (simulationtimes % 10 == 0)
        {
            if (vis != null)
            {
                vis.Exit();
            }
            vis = new VissimClass();
            vis.ShowMinimized();
            simulationtimes = 1;
        }
        run_VISSIM(vis, VISSIMFilePath_str, VISSIMLayoutPath_str, traveltimepath, dr, DrivingBHNumber, str_sep);
        simulationtimes++;
    }
}

public static void run_VISSIM(IVissim vis, string VISSIMFilePath_str, string VISSIMLayoutPath_str, string traveltimepath, DataRow dr, int DrivingBHNumber, string str_sep)
{
    try
    {
        vis.ShowMinimized();
        ChangeParameteroutVISSIM(VISSIMFilePath_str, dr, DrivingBHNumber); //change the parameters value from the trajectories
        vis.LoadNet(VISSIMFilePath_str, 0);
        vis.LoadLayout(VISSIMLayoutPath_str);
        ChangeParameterinVISSIM(vis, dr);
        Simulation sim = vis.Simulation;
        sim.RunContinuous();

        string[] TravelTimeStrArray = COPY_TravelTime(VISSIMFilePath_str, str_sep);
        string traveltime = TravelTimeStrArray[0];
        if (!File.Exists(traveltimepath))
        {
            string sectionIndex = TravelTimeStrArray[1];
            File.AppendAllText(traveltimepath, sectionIndex + Environment.NewLine);
        }
        File.AppendAllText(traveltimepath, traveltime + Environment.NewLine);
    }
    catch (Exception e)
    {
        try
        {
            if (vis != null)
            {
                vis.Exit();
            }
        }
        catch (Exception error)
        {
        }
        vis = new VissimClass();
        run_VISSIM(vis, VISSIMFilePath_str, VISSIMLayoutPath_str, traveltimepath, dr, DrivingBHNumber, str_sep);
    }
}
//change Average standstill distance, Additive part of desired safety distance
//Multiplicative part of desired safety distance, Lane change distance and
//Emergency stop distance
public static void ChangeParameterinVISSIM(IVisSim vis, DataRow dr)
{
    // Average standstill distance, Additive part of desired safety distance, Multiplicative part of desired safety distance,
    // Lane change distance and Emergency Stop distance are changed with this function. dr is the parameters value
    // from the optimized EE trajectories
    IDrivingBehaviorParSets dbps = vis.Net.DrivingBehaviorParSets;
    foreach (IDrivingBehaviorParSet dbp in dbps)
    {
        dbp.set_AttValue("AXADD", dr["Average standstill distance"]) ;
        dbp.set_AttValue("BXADD", dr["Additive part of desired safety distance"]) ;
        dbp.set_AttValue("BXMULT", dr["Multiplicative part of desired safety distance"]) ;
    }

    ILinks links = vis.Net.Links;
    foreach (ILink link in links)
    {
        object o = link.get_AttValue("CONNECTOR");
        if (Convert.ToInt32(o) == -1)
        {
            link.set_AttValue("LANECHANGEDISTANCE", dr["Lane change distance"]) ;
            link.set_AttValue("EMERGENCYSTOP", dr["Emergency stop distance"]) ;
        }
    }
}

public static void ChangeParameteroutVISSIM(string VISSIMFilePath_str, DataRow dr, int DrivingBHNumber)
{
    // This function is used to change the parameters value which can not be modified by the vissim com functions.
    // The method is by changing the value directly in the input file through file I/O operations
    int startrow = DrivingBHNumber + 5;
    string[] Lines = File.ReadAllLines(VISSIMFilePath_str);
    string[] str = PrintVISSIM_Input(dr);
    for (int i = 0; i <= 4; i++)
    {
        Lines[startrow + i * 18] = str[0];
        Lines[startrow + 1 + i * 18] = str[1];
        Lines[startrow + 2 + i * 18] = str[2];
        Lines[startrow + 3 + i * 18] = str[3];
    }
    File.WriteAllLines(VISSIMFilePath_str, Lines);
}

public static string[] PrintVISSIM_Input(DataRow dr)
{
    string[] str = new string[4];
    str[0] = " T_DISAPPEAR 60.00 MIN_LC_GAP " + dr["Minimum headway"] + " MIN_FREEFLOW 0.00";
    str[1] = " MIN_ACCELERATION OWN MIN " + dr["Max deceleration (own)"] + " DISTANCE " + dr["-1 m/s^2 per distance (own)"] + " MAX " + dr["Accepted deceleration (own)"];
    str[2] = " TRAILING_VEHICLE MIN " + dr["Max deceleration (trailing)"] + " DISTANCE " + dr["-1 m/s^2 per distance (trailing)"] + " MAX " + dr["Accepted deceleration (trailing)"];
    str[3] = " COOPERATIVE " + dr["Max. deceleration for cooperative braking"] + " LOOKAHEAD ABXFACTOR " + dr["Safety distance reduction factor"];
    return str;
}

public static System.Data.DataTable ReadCSVtoDataTable(string CSVDataPath_str)
{
    string[] Lines = File.ReadAllLines(CSVDataPath_str);
    string[] Fields;
    char sep_str;
    if (Lines[0].IndexOf(',') > -1)
    {
        sep_str = ',';
    }
    else
    {
        sep_str = ';';
        Fields = Lines[0].Split(new char[] { sep_str });
    }
    int Cols = Fields.GetLength(0);
    System.Data.DataTable dt = new System.Data.DataTable();
    // int row must be column names; force lower case to ensure matching later on.
    for (int i = 0; i < Cols; i++)
    {
        dt.Columns.Add(Fields[i].ToLower(), typeof(string));
    }
    for (int i = 1; i < Lines.GetLength(0); i++)
    {
        Fields = Lines[i].Split(new char[] { sep_str });
        Row = dt.NewRow();
        for (int f = 0; f < Cols; f++)
        {
            double d = Double.Parse(Fields[f]);
            Row[f] = String.Format("(0:02)\", d);
        }
        dt.Rows.Add(Row);
    }
    return dt;
}

public static string COPY_NET_TravelTime(string path)
{
    int length = path.Length;
    string traveltimepath = path.Remove(length - 4) + ".npe";
    string[] Lines = File.ReadAllLines(traveltimepath);
    int row = Lines.GetLength(0);
    string line = Lines[row - 1];
}
public static string[] COPY_TravelTime(string path, string str_sep)
{
    int length = path.Length;
    string traveltimepath = path.Remove(length - 4) + ".rsz";
    string[] Lines = File.ReadAllLines(traveltimepath);
    int row = Lines.GetLength(0);
    string line = Lines[row - 1];
    line = line.Replace(" ", "");
    string[] str_array = line.Split(new char[] { ';' });
    int str_length = str_array.Length;
    StringBuilder builder = new StringBuilder();
    StringBuilder builder2 = new StringBuilder();
    for (int i = 0; 2 * i + 1 < str_length - 1; i++)
    {
        builder.Append(str_array[2 * i + 1]);
        builder.Append(str_sep);
        builder2.Append("Section-" + (i + 1).ToString());
        if (2 * i + 3 < str_length - 1)
        {
            builder.Append(str_array[2 * i + 2]);
            builder.Append(str_sep);
            builder2.Append(str_sep);
        }
    }
    string[] return_linearray = new string[2];
    return_linearray[0] = builder.ToString();
    return return_linearray;
}